



**CENTRO UNIVERSITÁRIO DE BRASÍLIA**  
**FACULDADE DE TECNOLOGIA E CIÊNCIAS SOCIAIS APLICADAS**  
**CURSO DE ENGENHARIA DE COMPUTAÇÃO**

**ANDRÉ JOSÉ MARTINS DE SOUZA**

**CONTROLE E AUTOMAÇÃO DE ILUMINAÇÃO RESIDENCIAL POR TELEFONE**

**Orientadora: Prof<sup>a</sup>. M.C. Maria Marony Sousa Farias**

**BRASÍLIA**

**DEZEMBRO, 2011**

**ANDRÉ JOSÉ MARTINS DE SOUZA**

**CONTROLE E AUTOMAÇÃO DE ILUMINAÇÃO RESIDENCIAL POR TELEFONE**

Trabalho apresentada ao UniCEUB – Centro  
Universitário de Brasília como pré-requisito para  
obtenção de Certificação de Conclusão do  
Curso de Engenharia de Computação.  
Orientadora: Prof<sup>a</sup>. Maria Marony Sousa Farias.

**BRASÍLIA**

**DEZEMBRO, 2011**

**ANDRÉ JOSÉ MARTINS DE SOUZA**

**CONTROLE E AUTOMAÇÃO DE ILUMINAÇÃO RESIDENCIAL POR TELEFONE**

Trabalho apresentada ao UniCEUB – Centro  
Universitário de Brasília como pré-requisito para  
obtenção de Certificação de Conclusão do  
Curso de Engenharia de Computação.  
Orientadora: Prof.<sup>a</sup> Maria Marony Sousa Farias.

**Este Trabalho foi julgado adequado para a obtenção do Título de Engenheiro  
de Computação, e aprovado em sua forma final pela Faculdade de Tecnologia  
e Ciências Sociais Aplicadas - FATECS.**

---

**Prof. Abiezer Amarília Fernandez**  
**Coordenador do Curso**

**Banca Examinadora:**

---

**Prof.<sup>a</sup> Maria Marony Sousa Farias,**  
**mestre em Engenharia Elétrica.**  
**Orientadora**

---

**Prof. Antônio Barbosa Júnior, espe-**  
**cialista em Engenharia de Software.**  
**UniCEUB**

---

**Prof. José Julimá Bezerra Júnior,**  
**mestre em Engenharia Elétrica.**  
**UniCEUB**

---

**Prof. Sávio Wanderley do Ó, mestre**  
**em Estrutura e Construção Civil.**  
**UniCEUB**

*Dedico a todos que de certa maneira  
me ajudaram a concluir mais esta fase da  
minha vida.*

## **AGRADECIMENTOS**

A minha família, pela dedicação ao longo da minha vida, sempre buscando as melhores oportunidades para mim e para meu irmão.

A minha namorada Aylanne Rezende, por estar ao meu lado me dando apoio e força nas horas certas.

A todos os colegas com quem estudei que contribuíram direta ou indiretamente para o desenvolvimento deste trabalho.

E a todos aqueles que um dia eu tenha comentado sobre meu projeto e que se mostraram empolgados, fazendo com que eu mantivesse o foco principal.

*“Nossa maior fraqueza é a desistência. O caminho mais certo para o sucesso é sempre tentar apenas uma vez mais.”*

**Thomas Edison**

## RESUMO

Este trabalho apresenta um dispositivo de controle e automação de iluminação residencial por telefone. O projeto integra um módulo GSM, microcontrolador PIC16F877A, sensor de luminosidade e relés, o projeto liga ou desliga a iluminação através de acionamento pelo sensor de luminosidade ou de acordo com a mensagem SMS padrão enviada pelo cliente previamente cadastrado. Controlado através de linguagem de programação C e com utilização de comandos AT, o microcontrolador faz a leitura do SMS ou da tensão do sensor LDR e age de acordo com o programado, enviando sinais lógicos de 0V ou 5V à placa de relés. O sistema tem uma logística de notificação ao cliente sobre quaisquer alterações de iluminação, seja pelo sensor LDR, seja por solicitação através de SMS do cliente. O protótipo é composto por maquete de uma casa em conjunto com o dispositivo, e mostrou resultados satisfatórios sendo possível sua replicação em uma área de atuação real.

Palavras-chave: Automação, Iluminação, GSM, PIC16F877A, C, Sensor LDR, Comandos AT, Mensagem SMS.

## **ABSTRACT**

This work features device automation and control home lighting by phone. The project integrates a GSM module, PIC16F877A microcontroller, lightness sensor and relays, the project ON or OFF lighting through the activation by lightness sensor or according to the standard SMS message sent by the customer previously registered. Controlled by programming in C language and use of AT commands, the microcontroller reads the SMS or voltage LDR and age in accordance with programmed, sending logic signals from 0V or 5V to relay board. The system has a logistics notification to the customer of any change of lighting, either by LDR, either by the request through SMS customer. The prototype consists by maquette of a home in conjunction with the device, and showed satisfactory results and possible replication in a real operation area.

Keywords: Automation, Lighting, GSM, PIC16F877A, C, LDR, AT Commands, SMS message.



## SUMÁRIO

<b>LISTA DE FIGURAS .....</b>	<b>XII</b>
<b>LISTA DE TABELAS .....</b>	<b>XIV</b>
<b>LISTA DE ABREVIATURAS E SIGLAS.....</b>	<b>XV</b>
<b>CAPÍTULO 1 - INTRODUÇÃO .....</b>	<b>16</b>
1.1 Introdução ao Tema Proposto.....	16
1.2 Motivação .....	16
1.3 Objetivos .....	17
1.4 Metodologias .....	17
1.5 Estrutura da Monografia .....	18
<b>CAPÍTULO 2 – APRESENTAÇÃO DO PROBLEMA.....</b>	<b>19</b>
2.1 Furtos a Residências .....	19
2.2 Segurança Residencial .....	19
2.3 Conforto Residencial .....	20
2.4 Soluções Existentes.....	20
2.5 Benefícios do Dispositivo Proposto e Suas Restrições .....	22
<b>CAPÍTULO 3 – REFERENCIAL TEÓRICO .....</b>	<b>23</b>
3.1 Automação Residencial .....	23
3.2 Iluminação Residencial .....	24
3.3 Microcontroladores .....	25

3.3.1	Microcontroladores da família PIC .....	26
<b>3.4</b>	<b>Telefonia Móvel.....</b>	<b>27</b>
3.4.1	Rede GSM .....	27
3.4.2	SMS ( <i>Short Message Service</i> ).....	28
3.4.3	SIM Card.....	30
<b>3.5</b>	<b>Visão Geral do Projeto .....</b>	<b>31</b>
<b>CAPÍTULO 4 – DESCRIÇÃO DO HARDWARE E SOFTWARE .....</b>		<b>32</b>
<b>4.1</b>	<b>Microcontrolador PIC 16F877A .....</b>	<b>32</b>
4.1.1	Especificações .....	32
4.1.2	Pinagem do PIC 16F877A .....	33
4.1.3	CCS C Compiler .....	36
4.1.4	Kit de Gravação PICKit 2.....	36
4.1.4.1	Utilização do Kit de Gravação PICKit 2 .....	37
<b>4.2</b>	<b>Kit de desenvolvimento para SIM900D .....</b>	<b>38</b>
4.2.1	Especificações do módulo GSM, SIM900D .....	39
4.2.2	Comandos AT .....	40
4.2.3	HyperTerminal .....	40
<b>4.3</b>	<b>Comunicação Serial RS-232 .....</b>	<b>41</b>
4.3.1	Circuito Integrado MAX232 .....	42
4.3.2	Pinagem do CI MAX232 .....	42
<b>4.4</b>	<b>Teclado Numérico .....</b>	<b>44</b>
<b>4.5</b>	<b>Módulo LCD .....</b>	<b>45</b>
<b>4.6</b>	<b>Sensor LDR (<i>Light Dependent Resistor</i>) .....</b>	<b>46</b>
<b>4.7</b>	<b>Relé.....</b>	<b>47</b>
<b>CAPÍTULO 5 – IMPLEMENTAÇÃO.....</b>		<b>48</b>
<b>5.1</b>	<b>Modelagem do Sistema.....</b>	<b>49</b>

5.1.1 – Fluxograma Geral do Sistema .....	50
<b>5.2 Elaboração dos Circuitos .....</b>	<b>50</b>
5.2.1 Proteus ISIS 7 Professional .....	51
5.2.2 Elaboração dos Circuitos na Matriz de Contato .....	53
<b>5.3 Elaboração do Código Fonte para o Microcontrolador PIC .....</b>	<b>55</b>
5.3.1 Escrita do Código Fonte .....	55
5.3.2 Compilação e Gravação do Código Fonte .....	60
<b>5.4 Montagem dos Circuitos nas Placas .....</b>	<b>61</b>
<b>5.5 Montagem do Protótipo .....</b>	<b>63</b>
<b>CAPÍTULO 6 – RESULTADOS OBTIDOS .....</b>	<b>66</b>
<b>6.1 Simulações.....</b>	<b>66</b>
6.1.1 Cadastro do número do cliente no sistema.....	67
6.1.2 Alteração da iluminação pelo sensor de LDR .....	70
6.1.3 Alteração da iluminação pelo cliente .....	72
<b>6.2 Problemas Encontrados .....</b>	<b>73</b>
<b>CAPÍTULO 7 – CONSIDERAÇÕES FINAIS .....</b>	<b>74</b>
<b>7.1 Conclusões .....</b>	<b>74</b>
<b>7.2 Propostas para Trabalhos Futuros .....</b>	<b>75</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>76</b>
<b>APÊNDICE A - CÓDIGO FONTE DO DISPOSITIVO.....</b>	<b>79</b>
<b>APÊNDICE B – CUSTO DOS COMPONENTES DO DISPOSITIVO .....</b>	<b>88</b>
<b>APÊNDICE C – CIRCUITOS DAS PLACAS DO DISPOSITIVO .....</b>	<b>89</b>

## LISTA DE FIGURAS

Figura 2.1 – Sensor de Presença.....	21
Figura 2.2 – Minuteiro. ....	21
Figura 2.3 – Dimmer.....	21
Figura 2.4 – Interruptor-Cartão.....	21
Figura 3.1 – Dimensionamento das famílias PIC e dsPIC.....	26
Figura 3.2 – Rede SMS de Celular.....	29
Figura 3.3 – Diagrama Esquemático do Projeto.....	31
Figura 4.1 – PIC 16F877A utilizado no projeto.....	32
Figura 4.2 – Pinagem do PIC16F877A.....	33
Figura 4.3 – Pinagem do PIC 16F877A utilizada no projeto.....	35
Figura 4.4 – Kit de Gravação PICkit 2 .....	36
Figura 4.5 – Tela de gravação do programa PICkit 2.....	37
Figura 4.6 – Kit de desenvolvimento e módulo SIM900D.....	38
Figura 4.7 – Conector DB-9 com sua pinagem. ....	41
Figura 4.8 – MAX232 utilizado no projeto .....	42
Figura 4.9 – Pinagem do MAX 232 .....	42
Figura 4.10 – Pinagem do MAX232 utilizada no projeto. ....	43
Figura 4.11 – Teclado Numérico e Diagrama do Circuito.....	44
Figura 4.12 – Interligação do teclado numérico utilizado no projeto.....	45
Figura 4.13 – Módulo LCD utilizado no projeto .....	45
Figura 4.14 – Sensor LDR utilizado no projeto e variação de resistência com a luz. ....	47
Figura 4.15 – Relés utilizados no projeto .....	47
Figura 5.1 – Protótipo na Fase Final .....	48
Figura 5.2 – Disposição dos Componentes e Planta Baixa da Casa .....	49
Figura 5.3 – Fluxograma Geral do Sistema.....	50
Figura 5.4 – Circuito da Placa Principal feita no Proteus ISIS.....	51
Figura 5.5 – Inserção do código fonte .....	52
Figura 5.6 – Circuito da Placa de Relés feita no Proteus ISIS .....	53

Figura 5.7 – Circuito da Placa Principal feita na Matriz de Contato.....	54
Figura 5.8 – Relé e diodo de proteção. ....	55
Figura 5.9 – Código fonte, diretivas de pré-compilação. ....	56
Figura 5.10 – Código fonte, chamada de funções e variáveis.....	57
Figura 5.11 – Código fonte, função para uso de interrupção. ....	57
Figura 5.12 – Código fonte, função de envio de mensagem SMS .....	58
Figura 5.13 – Código Fonte, laço principal do sistema.....	59
Figura 5.14 – Compilador do CCS C Compiler em execução. ....	60
Figura 5.15 – Desenho da Placa de Relés e Placa descobreada pronta. ....	61
Figura 5.16 – Placa Principal e Disposição dos Componentes. ....	62
Figura 5.17 – Placa de Relés .....	63
Figura 5.18 – Maquete da casa em fase de construção.....	63
Figura 5.19 – Bocal e lâmpada utilizados no projeto.....	64
Figura 5.20 – Caixa de Interruptores e Tomadas. ....	64
Figura 5.21 – Disposição dos componentes a frente da casa. ....	65
Figura 5.22 – Esquema das ligações elétricas. ....	65
Figura 6.1 – Teste nos Interruptores .....	66
Figura 6.2 – Teste nas Tomadas com Chave de Teste.....	67
Figura 6.3 – Protótipo – Mensagem Inicial do sistema.....	68
Figura 6.4 – Protótipo – Ligando o módulo GSM. ....	68
Figura 6.5 – Protótipo – Digitando o número do cliente .....	69
Figura 6.6 – Protótipo – Confirmando o número do cliente .....	69
Figura 6.7 – Protótipo – Mensagem final do sistema. ....	69
Figura 6.8 – Chave do Sensor LDR ligada. ....	70
Figura 6.9 – Protótipo – Testes feitos para o sensor LDR.....	71
Figura 6.10 – Alteração pelo cliente. Mensagem enviada pelo cliente (lado esquerdo), alteração no protótipo (no centro), mensagem recebida pelo cliente (lado direito) .....	72

**LISTA DE TABELAS**

Tabela 4.11– Significado das nomenclaturas dos pinos do PIC16F877A. ....	34
Tabela 4.22– Identificação de cada componente presente na Figura 4.6 .....	38
Tabela 4.33– Comandos AT utilizados no Projeto. ....	40
Tabela 4.44– Conector DB9 com direção e designação dos pinos. ....	41
Tabela 4.55– Níveis de tensão – lógica TTL e padrão RS-232 .....	42
Tabela 4.66– Pinagem do teclado numérico .....	44
Tabela 4.77– Pinos do módulo LCD e interfaceamento no projeto .....	46
Tabela 5.18– Descrição de diretivas do código fonte. ....	56

**LISTA DE ABREVIATURAS E SIGLAS**

<b><math>\Omega</math></b>	Ohms
<b>A</b>	Ampère
<b>A/D</b>	Analógico / Digital
<b>Anatel</b>	Agência Nacional de Telecomunicação
<b>AT</b>	Hayes AT Commands
<b>ASCII</b>	American Standard Code for Information Interchange
<b>Aureside</b>	Associação Brasileira de Automação Residencial
<b>bps</b>	bit por segundo
<b>DB9</b>	Data Bus 9
<b>EEPROM</b>	Erasable Electronically Programmable Read Only Memory
<b><math>\mu</math>F</b>	Micro-Farad
<b>GMSC</b>	Gateway Mobile Switching Center
<b>GSM</b>	Global System for Mobile Communications
<b>I2C</b>	Inter-Integrated Circuit
<b>IBGE</b>	Instituto Brasileiro de Geografia e Estatística
<b>LCD</b>	Liquid Cristal Display
<b>LDR</b>	Light Dependent Resistor
<b>LED</b>	Diodo Emissor de Luz
<b>MSC</b>	Mobile Switching Center
<b>PIC</b>	Programmable Interface Controller
<b>SIM</b>	Subscriber Identification Module
<b>SMS</b>	Short Message Service
<b>SMSC</b>	Short Message Service Center
<b>SPI</b>	Serial Peripheral Interface
<b>TCP/IP</b>	Transmission Control Protocol / Internet Protocol
<b>TTL</b>	Transistor-Transistor Logic
<b>USART</b>	Universal Synchronous Asynchronous Receiver Transmitter
<b>V</b>	Volts

## **CAPÍTULO 1 - INTRODUÇÃO**

### **1.1 Introdução ao Tema Proposto**

O controle e automação residencial têm aparecido como uma tendência no cenário mundial atual. Por apresentar um custo relativamente alto para a realidade da maioria da população o controle e automação residencial ainda estão muito distantes de assumir seu total potencial.

De acordo com Muratori<sup>1</sup> (2010), nos últimos três anos, o mercado de controle e automação residencial vem crescendo a uma média de 35% ao ano em número de projetos e seu custo está 50% mais barato do que há cinco anos.

Com o passar do tempo o número de pessoas que instalam algum tipo de dispositivo de controle e automação residencial vêm crescendo, seja para aumentar a segurança do seu lar ou para ter uma maior comodidade e não se preocupar com tarefas comuns, diminuindo assim o estresse do dia a dia.

### **1.2 Motivação**

Na área de controle e automação de residências, um dos principais ramos é a iluminação residencial. Muitos dispositivos, atualmente, relacionados à iluminação residencial têm a função de coibir indivíduos mal intencionados, são dispositivos que com a aproximação do indivíduo fazem com as luzes da localidade se acendam.

Há também preocupações menores, como por exemplo, pessoas que costumam trabalhar o dia todo, saindo pela manhã e retornando apenas a noite aos seus lares. Deixar a iluminação da sala ou da entrada da casa acesa o dia todo seria uma das soluções para aparentar que a casa está habitada e ajudar ao proprietário, tanto para estacionar o carro na garagem, como para adentrar a casa sem ter o risco de esbarrar em algum objeto a frente, porém tal solução se tornaria inviável, e proporcionaria um aumento considerável na conta de energia da casa.

---

<sup>1</sup> José Roberto Muratori é engenheiro de produção com especialização em Administração de Empresas e foi membro fundador da Aureside. Atualmente participa da *Marbie Systems*.



Uma das soluções e que não geraria um gasto desnecessário de energia elétrica seria a instalação de uma central de controle e automação de iluminação residencial. Com isso a energia seria usada apenas quando necessária, sendo capaz de controlar e se programado previamente poderia automatizar a iluminação dos cômodos da casa, tais como: iluminação da garagem, iluminação da varanda, Iluminação da sala e qualquer outro cômodo que o proprietário quisesse ter o controle.

### **1.3 Objetivos**

O projeto tem como objetivo geral a elaboração de um dispositivo de controle e automação de iluminação residencial por telefone, utilizando para isso mensagens SMS.

O objetivo específico deste trabalho são:

- Apresentação de um protótipo de uma casa com iluminação dos cômodos controlada pelo dispositivo;
- Uso de microcontrolador em conjunto com um módulo GSM;
- Uso de sensor de luminosidade para acionamento da iluminação de alguns cômodos;
- Utilização de uma mensagem SMS padrão pelo cliente para alterar a iluminação dos cômodos da casa;
- Envio, pelo dispositivo, de mensagem SMS informando quaisquer alterações que ocorram na iluminação da casa, seja por solicitação do cliente, seja devido à variação da luminosidade natural no sensor de luminosidade.

### **1.4 Metodologias**

Para a elaboração deste projeto, foram realizadas pesquisas bibliográficas em bibliografia técnica especializada e sites conceituados da *internet*, bem como a realização de diversos testes para a análise dos componentes eletrônicos que foram utilizados no dispositivo.

## 1.5 Estrutura da Monografia

Esta monografia é dividida em sete capítulos, incluindo a INTRODUÇÃO, que trata da introdução ao tema proposto, a motivação do projeto, os principais objetivos, metodologias de elaboração e pesquisa. E essa seção que descreve toda a estrutura da monografia.

No segundo capítulo, APRESENTAÇÃO DO PROBLEMA, é apresentada uma descrição aprofundada sobre as questões motivacionais do trabalho, as soluções existem atualmente sobre o tema proposto. E a última seção deste capítulo aborda de forma resumida os benefícios do dispositivo proposto e suas restrições.

No terceiro capítulo, REFERENCIAL TEÓRICO, trata de assuntos como automação residencial, microcontroladores, telefonia móvel de forma conceitual. Nesse capítulo é apresentada também uma visão geral do projeto.

No quarto capítulo, DESCRIÇÃO DO HARDWARE E SOFTWARE, aborda as especificações dos dispositivos utilizados, é detalhada a especificação dos componentes de controle de comunicação serial, do microcontrolador PIC utilizado, detalhamento do sensor LDR e dos relés, bem como a configuração do kit de desenvolvimento do módulo GSM, dentre outros dispositivos e *softwares*.

No quinto capítulo, IMPLEMENTAÇÃO, são apresentados cinco etapas necessárias para compreensão geral da implementação do projeto.

No sexto capítulo, RESULTADOS OBTIDOS, abordam as simulações que tiveram como objetivo testar todas as funcionalidades propostas do dispositivo, de forma a simular um ambiente real, bem como, as dificuldades encontradas.

No sétimo capítulo, CONSIDERAÇÕES FINAIS, são apresentadas a conclusão e as sugestões para trabalhos futuros.

## CAPÍTULO 2 – APRESENTAÇÃO DO PROBLEMA

Este capítulo tem como finalidade detalhar questões motivacionais do trabalho, como: segurança e comodidade residencial, que foram apresentados resumidamente na seção — Motivação — do capítulo anterior.

Serão apresentadas também algumas soluções existentes para essas questões, no que diz respeito à iluminação residencial. E por fim, são apresentados os benefícios do dispositivo proposto pelo trabalho e suas restrições.

### 2.1 Furtos a Residências

O delegado José Carlos Medeiros, da 21ª Delegacia de Polícia (Taguatinga) alerta a população. “Evitem deixar suas casas vazias, instalem alarmes e não deixem objetos de valor dentro dos carros”, diz. Ele afirma que em **90% das vezes, os crimes são casuais**, “ou seja, quando a ocasião faz o ladrão”. (CORREIOWEB, 2011, grifo meu).

O parágrafo acima nos mostra a realidade dos dias atuais, e nos mostra, conforme grifo apresentado, que 90% das vezes, os crimes são casuais, isso quer dizer que se em uma área residencial com algumas dezenas de casas, o meliante com pretensão de invadir e furtar irá escolher aquela casa que estiver mais fácil de ser adentrada, ou seja, as casas que não apresentarem qualquer tipo de segurança.

Dentre essas casas, o meliante verificará também, qual aparenta estar desocupada. Com isso, as residências que tiverem alguma iluminação ligada serão as primeiras a serem descartadas de uma possível invasão, e possíveis furtos de objetos que ali estiverem.

### 2.2 Segurança Residencial

O local mais vulnerável, na questão de segurança pessoal, é a residência. E por isso, a cada dia que passa seus proprietários vêm se prevenindo, através de:

- Instalação de sistema eletrônico de segurança;
- Utilização de cães de guarda adestrados;

- Boa convivência com os vizinhos, o que gera um cuidado mútuo, no caso de algo errado esteja acontecendo com a residência do outro;
- Não divulgação de hábitos da família a pessoas estranhas;
- Protegendo aberturas no telhado e sistemas de exaustão;
- Mantendo a iluminação ligada, principalmente nas entradas.

Estas são algumas das iniciativas tomadas por proprietários, a fim de diminuir as chances de terem a residência invadida e furtada.

## **2.3 Comodidade Residencial**

Além da segurança residencial, os moradores também estão em busca de uma maior comodidade residencial. Ao sair pela manhã para o trabalho e retornar somente à noite, o morador quer que, quando chegar à sua casa, a luz da garagem já esteja acesa, para assim facilitar o estacionamento do carro.

Mas o morador não quer ter apenas o controle da iluminação da garagem, ele quer ter o controle da iluminação de outros cômodos, como cozinha, sala, quartos e qualquer outro cômodo que ele achar necessário.

Em busca desta comodidade e outras, e somada à evolução tecnológica surgiram os dispositivos que permitem o controle e a automação de rotinas e tarefas de uma casa, não só o controle da iluminação, como de muitos outros dispositivos, que serão apresentados na seção — Automação Residencial — do próximo capítulo.

## **2.4 Soluções Existentes**

Há no mercado uma infinidade de dispositivos para o uso em automação de iluminação, alguns dos mais comuns para o uso são: (FREITAS, 2010).

- Sensores de presença – são dispositivos que acionam o sistema de iluminação com circulação na ambiente. (Figura 2.1).
- Minuteiros – são equipamentos que desligam as lâmpadas sob seu comando após um tempo determinado. (Figura 2.2).



**Figura 2.1 – Sensor de Presença.**  
(FONTE: <http://www.cesel.com.br>)



**Figura 2.2 – Minuteiro.**  
(FONTE: <http://www.sumelnet.com>)

- *Dimmer* – é um regulador de luminosidade que utiliza recursos eletrônicos para fazer a luminosidade das lâmpadas variarem de zero a um máximo. (Figura 2.3).
- Célula fotoelétrica – utilizada para reduzir o consumo de energia em iluminação, atuando como sistema detector de luminosidade natural.
- Interruptor-cartão – controla a alimentação elétrica de um ou vários pontos mediante a colocação e/ou extração de um cartão plástico. (Figura 2.4).



**Figura 2.3 – Dimmer**  
(FONTE: [http:// www.alarmex.net](http://www.alarmex.net))



**Figura 2.4 – Interruptor-Cartão**  
(FONTE: <http:// www.simonbrasil.com.br>)

Dentre os dispositivos existentes temos também, alguns dispositivos não tão comuns e que têm devido sua complexidade tecnológica, o valor elevado. Dispositivo como a central inteligente, que concentra em apenas um aparelho todo o controle da iluminação da residência, é um exemplo.

## 2.5 Benefícios do Dispositivo Proposto e Suas Restrições

Esse projeto tem como finalidade apresentar um dispositivo de controle e automação de iluminação residencial por telefone, utilizando para isso equipamentos computacionais simples e de baixo custo.

Por ser um dispositivo que tem como foco o controle de iluminação utilizando o telefone, o maior benefício do dispositivo será a possibilidade de se ter total controle da iluminação dos cômodos ligados ao dispositivo por telefone e ter o *feedback*<sup>1</sup> do dispositivo, informando através de mensagem SMS o estado da iluminação dos cômodos, quando houver a alteração solicitada ou programada pelo cliente.

O projeto não tem como funcionalidade informar caso haja alguma alteração, via interruptor, da iluminação. Pensando nisso, quando o dispositivo estiver em funcionamento, temos o acionamento de um relé que tira o controle dos interruptores, ou seja, a iluminação só pode ser ligada ou desligada pelo dispositivo.

Por questão de melhor visualização e na tentativa de evitar possíveis erros de programação o projeto se restringe ao controle de iluminação de no máximo oito cômodos e foi projetado em um protótipo miniaturizado feito pelo autor.

---

<sup>1</sup> *Feedback* é o procedimento que consiste no provimento de informação à uma pessoa sobre uma ação executada por esta, objetivando informar sobre as ações executadas anteriormente.

## CAPÍTULO 3 – REFERENCIAL TEÓRICO

Para o melhor entendimento e desenvolvimento do projeto, é necessário o estudo de conceitos teóricos pertinentes ao projeto desenvolvido. Embora alguns dos conceitos citados demandem um detalhamento mais extenso do tema para sua total compreensão, visando manter o foco principal, apenas as características mais relacionadas ao projeto serão apresentadas.

Neste capítulo é apresentada também uma visão de forma geral do projeto proposto. Por isso é fundamental para o entendimento do projeto a compreensão dos tópicos abordados neste capítulo.

### 3.1 Automação Residencial

O objetivo da automação residencial é integrar iluminação, entretenimento, segurança, telecomunicações, aquecimento, ar condicionado e muito mais através de um sistema inteligente programável e centralizado. Como consequência fornece praticidade, segurança, conforto e economia para o dia a dia dos usuários. (ABREU, 2003).

A automação residencial pode ser aplicada em muitos dos dispositivos utilizados nas residências atualmente, tais como:

- a) Telefonia;
- b) Ar condicionado;
- c) Iluminação;
- d) *Home theater*;
- e) Eletrodomésticos;
- f) Vigilância e alarmes;
- g) Cortinas e portas automáticas.

No passado, o alto custo de instalação de equipamentos para automação residencial afastava mesmo quem tinha dinheiro para investir, tornando o uso de tecnologia dentro de casa uma excentricidade. Segundo estimativa da Associação

Brasileira de Automação Residencial<sup>1</sup> (Aureside), os preços de equipamentos e a instalação caíram pela metade nos últimos cinco anos. (LEAL, 2011).

Para Muratori (2010), o próprio nome — automação — pode ser considerado inadequado, pois a automação é apenas uma parte de um processo maior, que é a integração de todos os sistemas domésticos, o que inclui a instalação elétrica, a segurança do imóvel, a iluminação, toda rede de comunicações e as utilidades.

A automação residencial nos dias de hoje têm destacando a humanização dos projetos, fazendo com que correspondam exatamente ao que é deles esperado. Projetos que sabem superar oposições infundadas, transmitindo confiabilidade e privilegiando o uso transparente dos equipamentos escolhidos. (MURATORI, 2010).

### **3.2 Iluminação Residencial**

A iluminação residencial há muito tempo, deixou de ser apenas aquela lâmpada, acionada pelo interruptor instalado na parede. Atualmente, fabricantes oferecem requinte, conforto, exclusividade e automação em seus aparelhos.

Ambientes que dispõem do uso de controle remoto da iluminação local, permitindo a alteração de claridade, intensidade para diversas ocasiões é o que há de mais simples, no que diz respeito a novas tendências de iluminação residencial.

De acordo com Romano<sup>2</sup> (2011), a iluminação automatizada para uma residência, ocasiona o uso racional de cargas elétricas, o que, diminui o consumo. Outra forma de economia proporcionada pela automação da iluminação é o desligamento automático na ausência de pessoas.

Atualmente, além da automação e do controle local da iluminação, através de uma central instalada em algum ponto estratégico da residência, os fabricantes estão começando a oferecer o controle de iluminação residencial à dis-

---

<sup>1</sup> A Associação Brasileira de Automação Residencial (Aureside) tem como missão fomentar a adoção de tecnologias de automação residencial no país, homologar produtos e serviços na área, manter cursos de capacitação, formação e certificação profissional em automação residencial, divulgar artigos técnicos, realizar cursos e palestras na área, dentre outras missões.

<sup>2</sup> Rubens Augusto Romano é engenheiro eletricista e diretor da *Automatic House*.



tância. Com essa nova tecnologia o cliente não precisa mais estar na residência para ligar a iluminação de algum cômodo, basta que ele tenha um dispositivo móvel, como um celular, ou um *tablet*<sup>3</sup> que tenha acesso a central inteligente da residência.

### 3.3 Microcontroladores

O microcontrolador é um dispositivo semicondutor em forma de circuito integrado, que integra as partes básicas de um microcomputador - microprocessador, memórias não-voláteis e voláteis e portas de entrada e saída. Geralmente, é limitado em termos de quantidade de memória, principalmente no que diz respeito à memória de dados, é utilizada em aplicações específicas, ou seja, naquelas que não necessitam armazenar grandes quantidades de dados, como automação residencial, automação predial, automação industrial e automação embarcada. (GIMENEZ, 2005).

São vários os fornecedores de microcontroladores. Os principais, em termos de volume de vendas no Brasil, são: (PAIOTTI, 2009).

- Microchip Technology Inc. (<http://www.microchip.com>);
- Intel Corporation (<http://www.intel.com>);
- Atmel Corporation (<http://www.atmel.com>); e
- Texas Instruments (<http://www.ti.com>);

Dentre todos os microcontroladores, os mais utilizados em projetos acadêmicos, devido à facilidade de implementação e montagem em circuito estão:

- Microcontroladores da Família 8051 – lançada pela Intel, até hoje é conhecida e utilizada devido a sua facilidade de programação, em linguagem *assembly* graças ao seu poderoso conjunto de instruções;
- Arduino – não é um microcontrolador, e sim uma plataforma que foi criada para facilitar e baratear uso e programação de microcontroladores em seus circuitos mais complexos; e
- Microcontroladores da Família PIC, que serão abordados na próxima seção – Microcontroladores da família PIC.

---

<sup>3</sup> *Tablet* é um dispositivo em formato de prancheta que pode ser usado para acesso à Internet. (página na web: <http://www.mundodastribos.com/o-que-e-tablet-para-que-serve.html>)

### 3.3.1 Microcontroladores da família PIC

O microcontrolador PIC é um circuito integrado produzido pela *Microchip Technology Inc.*, que pertence à categoria dos microcontroladores, ou seja, um componente integrado que em um único dispositivo contém todos os circuitos necessários para realizar um completo sistema programável.

A grande vantagem da família PIC é que todos os modelos possuem um set de instruções bem parecido, assim como também mantêm muitas semelhanças entre suas características básicas. A utilização de um modelo torna a migração para outros modelos muito mais simples. (SOUZA, 2005)

A denominação família PIC é devido aos diversos microcontroladores fornecidos, que vão desde microcontroladores de 8-bits, passando por 16-bits, até os de 32-bits. A escolha vai depender da funcionalidade e desempenho requerido, conforme mostrada na Figura 3.1.

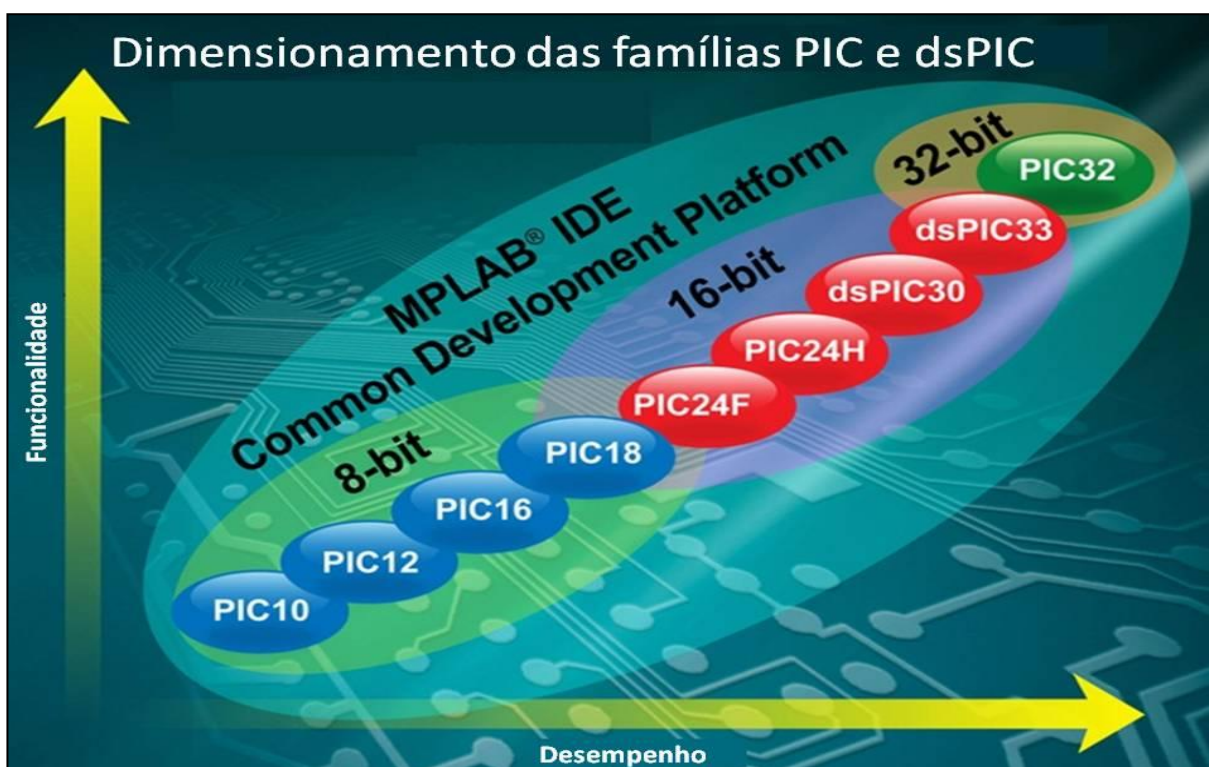


Figura 3.1 – Dimensionamento das famílias PIC e dsPIC.  
(Adaptado da página na web: <http://www.microchip.com>)

O microcontrolador da família PIC utilizado no projeto é o modelo 16F877A, devido à quantidade de portas de entrada e saída existentes, podendo assim acrescentar dispositivos, como um teclado numérico e levar o projeto a controlar a iluminação de uma maior quantidade de cômodos.

Detalhes referentes às principais características técnicas do microcontrolador PIC16F877A, serão abordadas na seção — Microcontrolador PIC16F877A — do próximo capítulo.

### 3.4 Telefonia Móvel

Devido ao trabalho proposto utilizar no seu funcionamento a telefonia móvel, no recebimento e envio de mensagem SMS, a noção teórica deste e outros assuntos relacionados são indispensáveis para entendimento do funcionamento do projeto.

Conforme divulgação feita em janeiro de 2011 pela Agência Nacional de Telecomunicações (Anatel), o Brasil terminou o ano de 2010 com um total de **202,94 milhões de telefones celulares**, que representa um crescimento de 16,66% em relação a 2009. (ÉPOCA NEGÓCIOS, 2011, grifo meu)

Como pode ser visto no parágrafo anterior o número de telefones celulares tem aumentado substancialmente de um ano para outro. A quantidade é tão surpreendente que passa o total da população brasileira, que no final de 2010 era de 190,7 milhões de habitantes, segundo dados obtidos no *site* do Instituto Brasileiro de Geografia e Estatística (IBGE) (página na web: <http://www.ibge.gov.br>).

Esse aumento se justifica, devido ao aparelho celular ter se tornado item obrigatório, não só para a comunicação de voz, como para comunicação através de mensagens SMS, acesso a *internet*, dentre outras funcionalidades presentes nos aparelhos mais modernos.

#### 3.4.1 Rede GSM<sup>4</sup>

Com o crescimento da telefonia móvel durante os anos 80, vários telefones celulares analógicos foram desenvolvidos na Europa, sendo que cada país desenvolvia seu próprio sistema, o que levou a incompatibilidades, devido à forma de envio de dados, protocolos e frequência de comunicação.

---

<sup>4</sup> GSM é a sigla em inglês de *Global System for Mobile Communications* (Sistema Global para Comunicações Móveis), antiga *Group Special Mobile*.

Com o objetivo de acabar com esses problemas, os Europeus criaram um grupo de estudos para desenvolver um sistema móvel, a fim de padronizar o sistema na Europa, o *Group Special Mobile* (GSM). Em 1989, os primeiros testes foram publicados, e no começo de 1994 já havia 1,3 milhões de assinantes em todo o mundo.

A rede GSM é uma tecnologia digital para celulares usada para transmissão de voz e serviços de dado móvel. Suporta chamadas de voz e dados como, por exemplo, o envio de mensagem SMS, com velocidade de transferência de até 9,6 kbps. (GSM WORLD, 2011, minha tradução).

### **3.4.2 SMS (*Short Message Service*)**

O Serviço de Mensagens Curtas (*Short Message Service*) permite aos usuários enviarem e receberem mensagens a partir de um aparelho de telefone celular. Cada mensagem pode ser de até 160 caracteres e ser enviada e recebida por usuários de diferentes redes de operadoras. (CAMPOS, 2010)

O SMS foi criado no final da década de 80 para funcionar com a tecnologia digital GSM. O principal objetivo era um sistema de mensagens bem simples que funcionasse mesmo quando os aparelhos dos usuários estivessem desligados ou fora da área de cobertura.

Alguns exemplos de utilização de serviços SMS são: (BERNAL, 2002)

- Mensagem ponto a ponto: este é o tipo mais comum de uso de SMS. A troca de mensagens SMS entre dois usuários móveis.
- Informação e serviços: citações de preços de ações, previsões do tempo e atualizações de notícias.
- Propagandas: envio de alertas a um usuário que se inscreveria para recebê-las. Pode ser utilizada como uma forma de propaganda.
- Atendimento ao consumidor: ferramenta de atendimento eficiente e barata para atender e prover informações a clientes.
- Despacho de trabalho: usado em despachos de encomendas e mensageiros em campo. Pode ser integrada com outras aplicações, como a de viaturas em trânsito.

### 3.4.2.1 Topologia da Rede SMS de celular

Na Figura 3.2 são mostrados os diversos componentes que compõem a rede SMS de celular. Observe que há uma porta de ligação que dá acesso móvel a *Internet* através de pilhas TCP/IP.

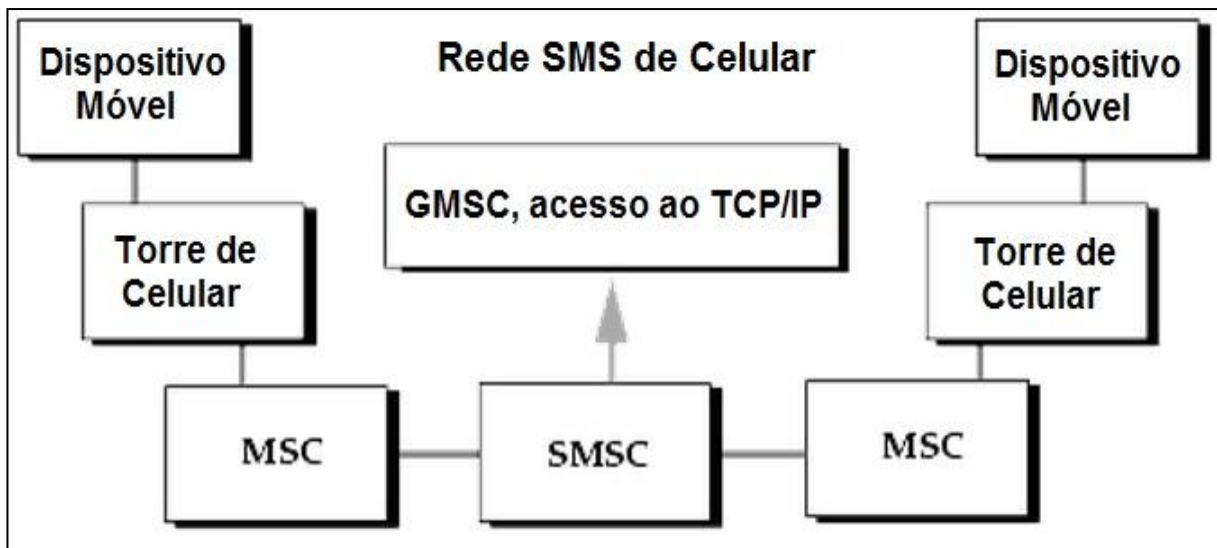


Figura 3.2 – Rede SMS de Celular  
(Adaptado da página na web: <http://developers.sun.com/mobility/midp/articles/sms/>)

Torre de celular – responsável pela transmissão de tráfego de voz e dados entre os dispositivos móveis e o MSC (*Mobile Switching Center*).

MSC (*Mobile Switching Center* – Central de Comutação Móvel) – coordena e controla a configuração de roteamento de chamadas entre telefones móveis em uma área de serviço. Possuem uma grande similaridade com as centrais de telecomunicação utilizadas na telefonia fixa.

SMSC (*Short Message Service Center* – Central de Serviço de Mensagens Curtas) – atua como um sistema de armazenamento e envio para a transmissão de mensagens de texto. O SMSC armazena as mensagens na rede até o telefone celular de destino fique disponível, garantindo assim a entrega da mensagem de texto pela rede.

GMSC (*Gateway Mobile Switching Center*) – é parecido com uma MSC, capaz de receber mensagens curtas de um SMSC, com a função adicional de interconectividade com outras redes de telecomunicações via redes TCP/IP.

### 3.4.2.2 Benefícios do SMS

De acordo com Bernal (2002) alguns dos benefícios do SMS são:

- Garantia de entrega pela notificação e alerta;
- Recebe mensagens durante o uso de serviços de voz e dados;
- Baixo custo e mecanismo de comunicação seguro para entrega de informação;
- Integração com aplicações baseadas no modelo da Internet, como, por exemplo, e-mail;
- Fonte de receita complementar para os provedores de serviço, tornando-se um modelo de negócios.

### 3.4.3 SIM<sup>5</sup> Card

O SIM *Card* ou Cartão SIM tem como função principal o armazenamento do número do telefone, com isso para a troca da linha telefônica para outro aparelho, basta colocar o cartão SIM no novo celular. Com o surgimento da tecnologia móvel GSM, que utiliza o cartão SIM na forma de um *chip*, o cartão SIM ganhou sucesso, e se tornou mais eficiente, barato e menor.

A memória do cartão SIM é do tipo EEPROM<sup>6</sup>, atualmente existem cartões SIM de diversos tamanhos, nela é armazenada a configuração básica do cartão SIM (cerca de 8 KB do espaço de memória) e o espaço restante é para aplicações adicionais, como por exemplo, armazenamento de mensagens SMS e agenda.

O cartão SIM da operadora A utilizado tem 64 KB de memória e 25 mm por 15 mm de tamanho.

---

<sup>5</sup> SIM é a sigla de *Subscriber Identity Module* (Módulo de Identificação do Assinante).

<sup>6</sup> EEPROM (*Electrically-Erasable Programmable Read-Only Memory*) – é um *chip* de armazenamento não-volátil, podendo ser programada e apagada várias vezes, eletricamente.

### 3.5 Visão Geral do Projeto

Após leitura e compreensão dos tópicos abordados neste capítulo, já é possível o entendimento teórico do projeto proposto. O projeto basicamente é constituído por módulo GSM (*kit* de desenvolvimento para SIM900D), microcontrolador (PIC do modelo 16F877A), conversor de sinais de porta serial (circuito Integrado MAX232), relés de duas posições, sensor LDR, teclado numérico, lâmpadas comuns de 220 V e demais componentes eletrônicos, como resistores, capacitores, oscilador de cristal, dentre outros.

O diagrama esquemático do projeto proposto pode ser observado na Figura 3.3, na qual representa de forma objetiva a composição geral do projeto.

Detalhes sobre o *hardware* e o *software* utilizados e seus funcionamentos encontram-se no capítulo 4 – Descrição de Hardware e Software. A implementação do projeto pode ser observada no capítulo 5 – Implementação.

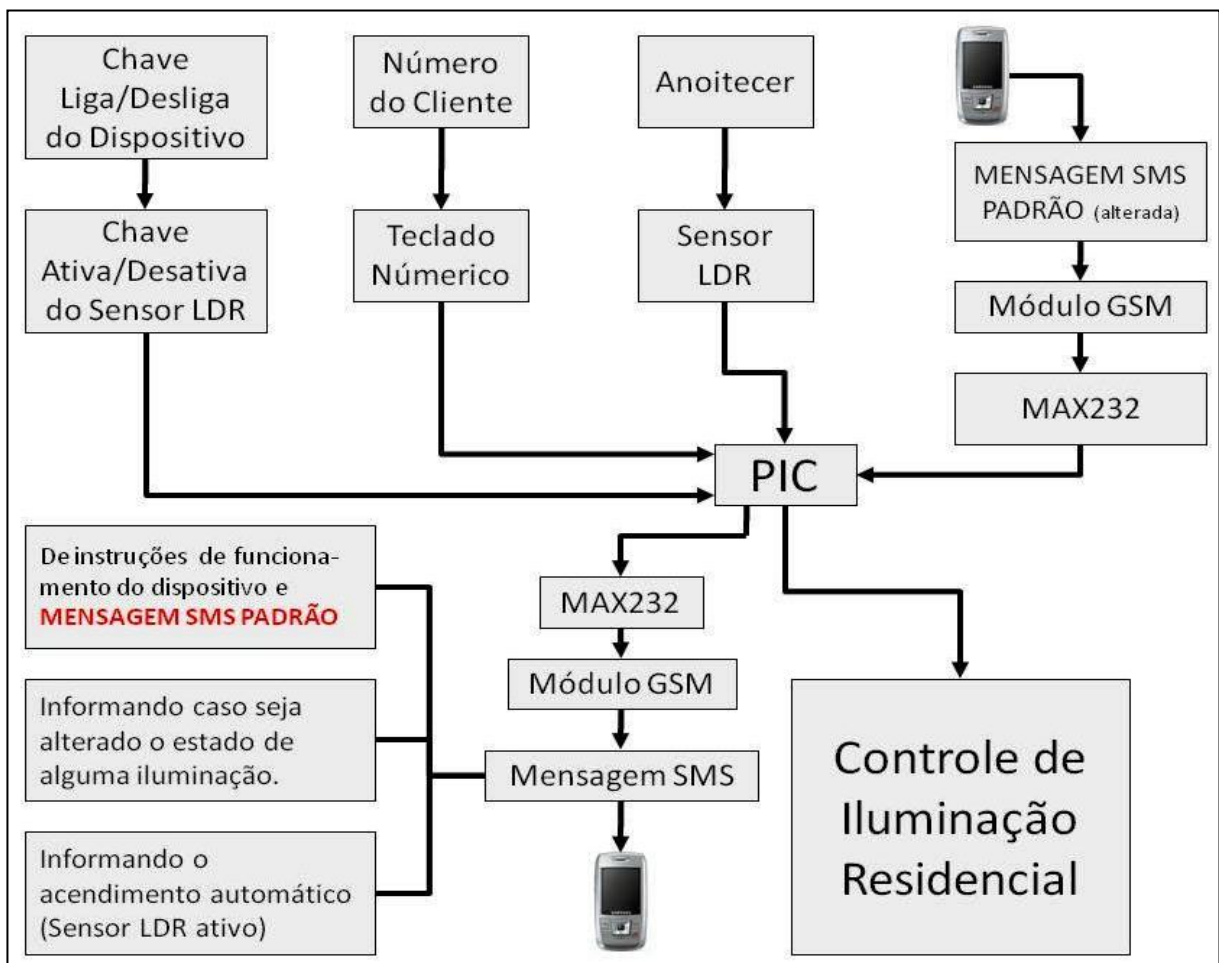


Figura 3.3 – Diagrama Esquemático do Projeto (AUTOR)

## CAPÍTULO 4 – DESCRIÇÃO DO HARDWARE E SOFTWARE

Este capítulo aborda as especificações, de forma detalhada, dos dispositivos utilizados e seu funcionamento neste projeto, tanto a parte física, ou seja, o *hardware*, como a parte lógica, os *softwares* utilizados.

### 4.1 Microcontrolador PIC 16F877A

Conforme mencionado no capítulo anterior, o microcontrolador utilizado nesse projeto foi o PIC 16F877A. É um modelo de 40 pinos, o que possibilita a montagem de um *hardware* complexo e capaz de interagir com diversos recursos ao mesmo tempo. A Figura 4.1 ilustra o PIC 16F877A utilizado neste projeto.

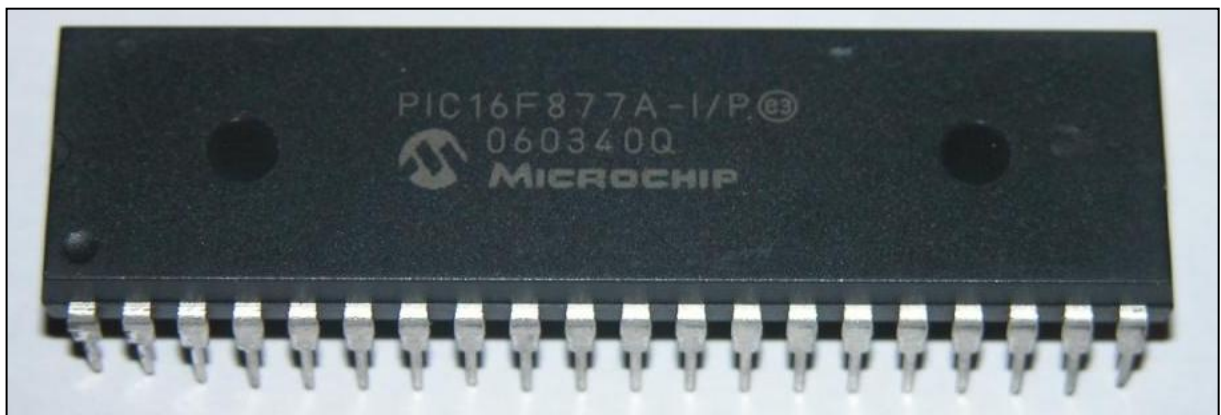


Figura 4.1 – PIC 16F877A utilizado no projeto (AUTOR)

#### 4.1.1 Especificações

As principais especificações do PIC 16F877A são: (SOUZA, 2005)

- Microcontrolador de 40 pinos;
- Via de programação com 14 bits e 35 instruções;
- Programação *in-circuit* (alta e baixa tensão);
- Frequência de operação de até 20 MHz;
- Memória *flash* de programa de 8kwords;
- 33 portas configuráveis como entrada ou saída;



- 15 interrupções disponíveis;
- Memória EEPROM interna com 256 bytes;
- Memória RAM com 368 bytes;
- Conversores A/D (analógico-digital) de 10 bits (8x) e comparadores analógicos (2x);
- Comunicações seriais: SPI, I2C e USART.

A escolha desse modelo em especial, se deve principalmente ao fato de possuir a grande quantidade de portas de entrada e saída, o modo de comunicação serial e conversores A/D, características essas que foram primordiais na elaboração deste dispositivo.

#### 4.1.2 Pinagem do PIC 16F877A

Na Figura 4.2 é mostrada a pinagem do PIC16F877A e a Tabela 4.1 o significado das nomenclaturas utilizadas na identificação desses pinos.

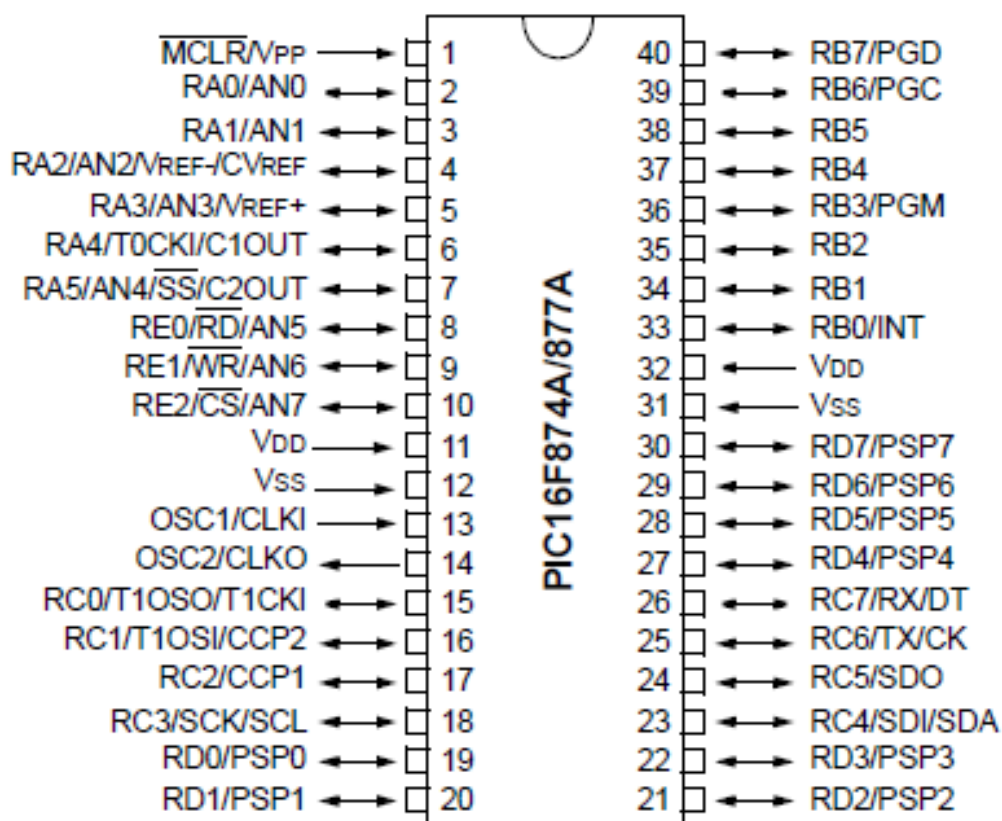


Figura 4.2– Pinagem do PIC16F877A  
(FONTE: Microchip Technology, 2003)

Tabela 4.1 – Significado das nomenclaturas dos pinos do PIC16F877A.

N°	Pino	Descrição
1	MCLR/Vpp	Master Clear (reset). O microcontrolador funciona quando este pino está em nível alto.
2	RA0 AN0	Entrada e saída digital. Entrada analógica.
3	RA1 AN1	Entrada e saída digital. Entrada analógica.
4	RA2 AN2 $V_{REF-}/CV_{REF}$	Entrada e saída digital. Entrada analógica. Tensão negativa de referência analógica.
5	RA3 AN3 $V_{REF+}$	Entrada e saída digital. Entrada analógica. Tensão positiva de referência analógica.
6	RA4 T0CKI C1OUT	Entrada e saída digital. Open-drain quando configurado como saída. Entrada externa do contador TMR0. Saída do comparador 1.
7	RA5 AN4 SS C2OUT	Entrada e saída digital. Entrada analógica. Slave para a comunicação SPI. Saída do comparador 2.
8	RE0 RD AN5	Entrada e saída digital. Controle de leitura da comunicação paralela. Entrada analógica.
9	RE1 WR AN6	Entrada e saída digital. Controle de escrita da comunicação paralela. Entrada analógica.
10	RE2 CS AN7	Entrada e saída digital. Habilitação externa para comunicação paralela. Entrada analógica.
11/32	VDD	Alimentação positiva.
12/31	VSS	GND.
13	OSC1/CLKIN	Oscilador cristal ou entrada de osciladores externos.
14	OSC2/CLKOUT	Saída para oscilador cristal.
15	RC0 T1OSO T1CKI	Entrada e saída digital. Saída do oscilador externo para TMR1. Entrada de incremento para TMR1.
16	RC1 T1OSI CCP2	Entrada e saída digital. Entrada do oscilador externo para TMR1. Entrada do Capture2 ou Saída para Compare2/PWM2.
17	RC2 CCP1	Entrada e saída digital. Entrada do Capture1 ou Saída para Compare1/PWM1.
18	RC3 SCK SCL	Entrada e saída digital. Entrada/Saída do clock para comunicação SPI. Entrada/Saída do clock para comunicação I2C.
19	RD0 PSP0	Entrada e saída digital. Comunicação paralela.
20	RD1 PSP1	Entrada e saída digital. Comunicação paralela.
21	RD2 PSP2	Entrada e saída digital. Comunicação paralela.
22	RD3 PSP3	Entrada e saída digital. Comunicação paralela.
23	RC4 SDI DAS	Entrada e saída digital. Entrada de dados para comunicação SPI. Entrada/Saída de dados para comunicação I2C.

24	RC5 SDO	Entrada e saída digital. Saída de dados para comunicação SPI.
25	RC6 TX CK	Entrada e saída digital. Transmissão para comunicação assíncrona USART. Clock para comunicação síncrona USART.
26	RC7 RX DT	Entrada e saída digital. Recepção para comunicação assíncrona USART. Dados para comunicação síncrona USART.
27	RD4 PSP4	Entrada e saída digital. Comunicação paralela.
28	RD5 PSP5	Entrada e saída digital. Comunicação paralela.
29	RD6 PSP6	Entrada e saída digital. Comunicação paralela.
30	RD7 PSP7	Entrada e saída digital. Comunicação paralela.
33	RB0 INT	Entrada e saída digital. Interrupção externa.
34	RB1	Entrada e saída digital.
35	RB2	Entrada e saída digital.
36	RB3 PGM	Entrada e saída digital. Entrada para programação de baixa tensão.
37	RB4	Entrada e saída digital.
38	RB5	Entrada e saída digital.
39	RB6 PGC	Entrada e saída digital. Clock de programação serial ou pino de in-circuit debugger.
40	RB7 PGD	Entrada e saída digital. Dado de programação serial ou pino de in-circuit debugger.

FONTE: SOUZA, 2005, p. 22.

A pinagem escolhida no projeto pode ser observada na Figura 4.3. Essa configuração atende às necessidades de cada componente utilizado no projeto.

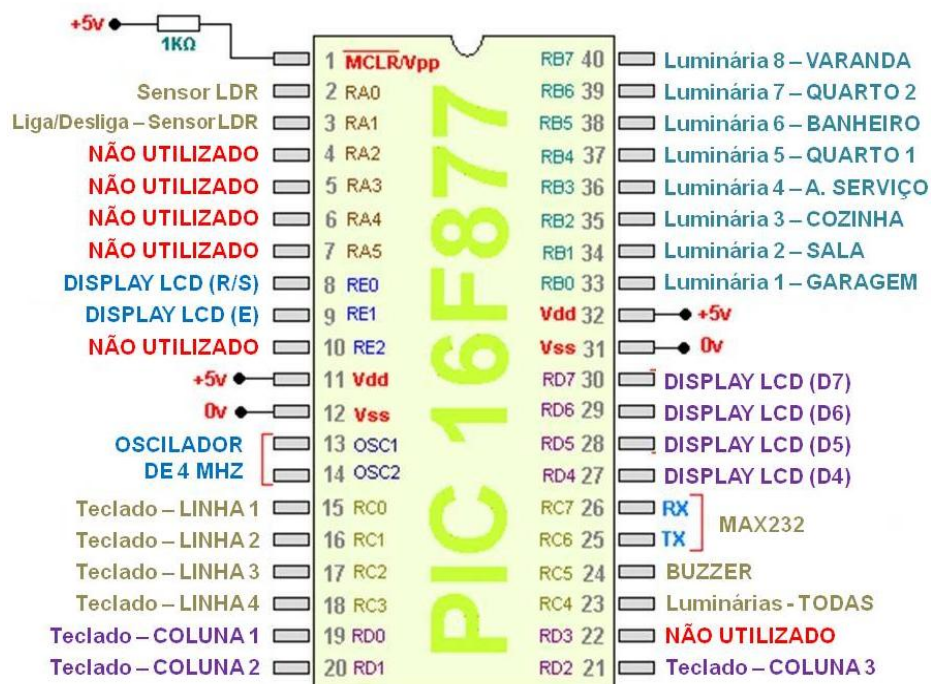


Figura 4.3 – Pinagem do PIC 16F877A utilizada no projeto. (AUTOR)

### 4.1.3 CCS C Compiler<sup>1</sup>

Este programa oferece uma suíte de ferramentas para o desenvolvimento e depuração de aplicativos embutidos em execução nos microcontroladores PIC. Assim, este programa foi utilizado para a compilação do código fonte desenvolvido na linguagem C, e que será melhor detalhada no próximo capítulo.

O CCS C *Compiler* compila códigos gravados com a extensão '.c'. Após compilação oito arquivos com o mesmo nome são gerados no mesmo local, com as seguintes extensões: '.cof', '.err', '.hex', '.lst', '.pjt', '.sta', '.sym', '.tre'. O programa além de compilar, mostra avisos e erros e a previsão do uso da memória RAM e ROM do microcontrolador. Após compilação, o arquivo '.hex' será utilizado, tanto na simulação no programa *Proteus*, como na gravação no microcontrolador PIC.

### 4.1.4 Kit de Gravação PICKit 2

O kit de gravação PICKit 2 é uma ferramenta de desenvolvimento que permite comunicar o computador diretamente com o PIC, foi adquirido pela empresa Robótica Simples (Página na Web: <http://roboticasimples.com>). É composto do conector ICSP (*In-Circuit Serial Programming*) para gravação em microcontroladores PIC de 8, 14, 18, 20, 28 e 40 pinos, conforme é mostrado na Figura 4.4.



**Figura 4.4 – Kit de Gravação PICKit 2 (AUTOR)**  
**Programa e cabo USB (lado esquerdo), Soquete (ao centro), Conector ICSP (lado direito)**

<sup>1</sup> O CCS C *Compiler* é um programa desenvolvido pela empresa *Custom Computer Service* (CCS), e está disponível para ser baixado na página na Web: <http://www.ccsinfo.com>.

Além da parte física, que contempla o gravador, o cabo USB, conector ICSP, o kit possui também um CD com o programa que tem o objetivo de identificar o microcontrolador a ser gravado e funciona em qualquer computador com uma porta USB disponível de 1.1 ou 2.0, conforme é mostrado na Figura 4.5.

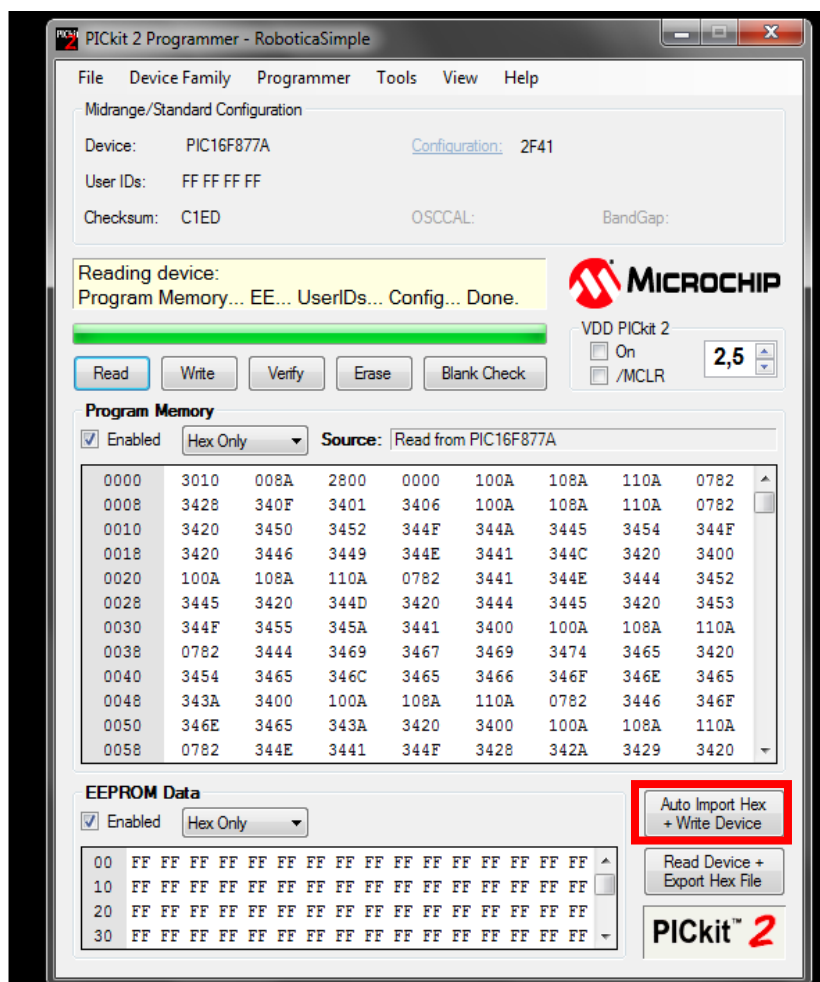


Figura 4.5 – Tela de gravação do programa PICkit 2. (AUTOR)

#### 4.1.4.1 Utilização do Kit de Gravação PICkit 2

De posse do arquivo gerado pelo CCS C Compiler, de extensão '.hex', o próximo passo foi abri-lo e gravá-lo no PIC 16F877A, utilizando o botão que se encontra destacado na Figura 4.5. Após localização do código fonte com a extensão '.hex' o processo de gravação será iniciado.

Assim que a gravação é completada o texto '*Programming Successful*' é mostrado na tela do programa e é emitido um sinal sonoro indicando que a gravação foi realizada com sucesso. Além da gravação a interface pode ser utilizada para apagar a memória do microcontrolador PIC.



## 4.2 Kit de desenvolvimento para SIM900D

O kit de desenvolvimento para SIM900D, nada mais é do que uma placa de desenvolvimento para utilização do módulo GSM denominado SIM900D e demais acessórios como: fonte de alimentação, antena, fone e cabo serial.

Na Figura 4.6 é mostrada a placa de desenvolvimento para SIM900D e o próprio módulo SIM900D e a Tabela 4.2 apresentam a identificação de cada componente presente na Figura 4.6.

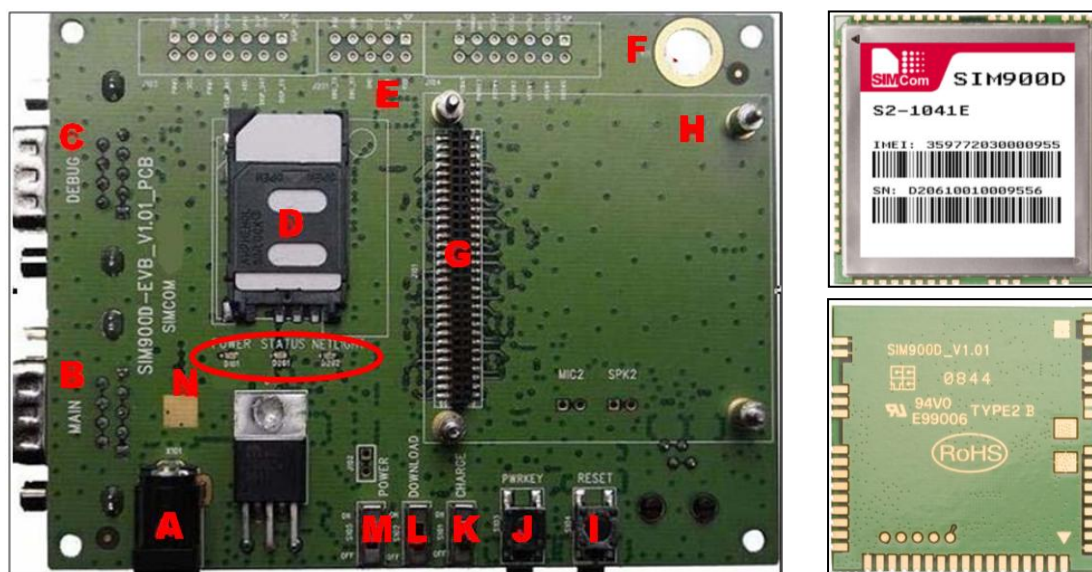


Figura 4.6– Kit de desenvolvimento e módulo SIM900D  
(FONTE: SIMCOM, 2010)

Tabela 4.2 – Identificação de cada componente presente na Figura 4.6

Letra	Descrição
A	Interface do Adaptador da Fonte.
B	Porta Serial Principal para transmissão dos Comandos AT.
C	Porta Serial Depuradora.
D	Interface do Cartão SIM.
E	Interface de ponto de teste.
F	Buraco de fixação da antena.
G	Interface de fixação do módulo SIM900D com o kit.
H	Área de fixação do módulo SIM900D.
I	Chave de Reset (no kit utilizado para este projeto não está presente).
J	Chave de Alimentação (Controle ON/OFF do módulo).
K	Interruptor de Charge (Controle ON/OFF Charge).
L	Interruptor de Download (Controle ON/OFF Download).
M	Interruptor de Alimentação (Controle ON/OFF da alimentação).
N	Led's indicativos de estado. (N1- Indica se o kit de desenvolvimento se encontra ligado; N2 – Indica o funcionamento normal do módulo GSM, se apagado, indica que o módulo funciona de forma não convencional; N3 – Pisca com certa frequência de acordo com estado da rede GSM.

FONTE: SIMCOM, 2010.

#### 4.2.1 Especificações do módulo GSM, SIM900D

O SIM900D é um módulo GSM *quad-band*, pode ser usado em aplicações onde a transmissão via tecnologia GSM se faça necessária; como transmissão de voz, envio de mensagens SMS. Consome pouca energia, tem seu tamanho reduzido que o torna ideal para tarefas Máquina a Máquina<sup>2</sup> (M2M) ou qualquer outra forma de comunicação móvel. O módulo SIM900D é fabricado pela SIMCOM<sup>3</sup> e distribuído no Brasil pela ME Componentes<sup>4</sup>.

As principais especificações técnicas do módulo SIM900D são: (ME COMPONENTES, 2011).

- Módulo de 48 pinos;
- *Quad-band* (850 MHz, 900 MHz, 1800 MHz e 1900 MHz);
- Compatível com tecnologia GSM;
- Pilha TCP/IP embutida;
- Dimensões: 33 mm por 33 mm por 3 mm;
- Peso: 6,2 gramas;
- Controle via comandos AT;
- Faixa de tensão de alimentação: 3,2 V a 4,8 V;
- Baixo consumo de energia: 1,0 mA (modo de espera);
- Temperatura de operação: -40° C a +85° C.

---

<sup>2</sup> Máquina a máquina (M2M) – é a utilização um dispositivo para capturar um evento que é retransmitido para um aplicativo que traduz o evento capturado em informações significativas.

<sup>3</sup> SIMCOM *Wireless Solutions* – é líder na indústria fornecendo módulos sem fio de alta qualidade para diferentes plataformas de tecnologias em redes. (Página na Web: <http://wm.sim.com>).

<sup>4</sup> ME Componentes e Equipamentos – é uma empresa fundada em 1997, que trabalha com todos os tipos e com as melhores marcas de componentes eletrônicos, tais como módulos GPS, módulos GSM/GPRS, circuitos integrados. (Página na Web: <http://www.mecomp.com.br>).

### 4.2.2 Comandos AT<sup>5</sup>

Os comandos AT são usados para configurar e solicitar alguns serviços nos módulos, como: fazer e receber discagem de chamadas de voz, enviar e receber SMS através de redes celulares de tecnologia GSM, dentre outros serviços.

Os comandos AT são um conjunto de linhas de comandos que são utilizadas para comunicação e configuração do módulo. O módulo GSM utilizado no projeto tem uma extensa lista de comandos AT que são reconhecidos, a Tabela 4.3 apresenta os comandos AT que foram utilizados no projeto e sua descrição.

**Tabela 4.3 – Comandos AT utilizados no Projeto.**

Comando	Descrição
AT	Verifica se há resposta do módulo
AT+CMGS	Envio de mensagem SMS
AT+CMGR	Leitura de mensagem SMS
AT+CMGDA	Apaga todas as mensagens SMS
AT+CSCS	Seleciona qual tipo de caractere será utilizado (ASCII)
AT+CMGF	Seleciona qual formato da mensagem SMS

FONTE: SIMCOM, 2010.

### 4.2.3 HyperTerminal

O programa *HyperTerminal* é o aplicativo de comunicações mais antigo do Microsoft Windows, atualmente apenas disponível na página oficial da Hilgraeve<sup>6</sup> (página na Web: <http://www.hilgraeve.com/hyperterminal>), gratuitamente por 30 dias.

Com ele é possível conectar o computador a outros sistemas remotos, utilizando para isso as portas seriais do computador. Este aplicativo foi utilizado na fase inicial para a realização de testes, onde foi possível através da digitação dos comandos AT que seriam utilizados visualizar o normal funcionamento do módulo GSM.

<sup>5</sup> Comandos AT – em inglês chamados de *Hayes AT Commands*, criados originalmente para a *Hayes Smartmodem* em 1981, por Dennis C. Hayes.

<sup>6</sup> Hilgraeve – Empresa privada de software, e é mais conhecida pelos seus programas: *HyperTerminal Private Edition* e *HyperAccess*, foi fundada em 1980.



### 4.3 Comunicação Serial RS-232

O padrão RS-232 define características mecânicas, elétricas e funcionais para a comunicação serial de dados entre um equipamento de dados terminal (computador, impressora) e um equipamento de comunicação de dados (modem). A comunicação entre o microcontrolador PIC16F877A e o módulo GSM SIM900D ocorreu utilizando o conector de nove pinos (DB9). Estes estão ligados ao MAX232 para que possam se comunicar. (ZANCO, 2010).

Na Figura 4.7 é mostrado o conector DB-9 com sua pinagem enumerada, enquanto o Tabela 4.4 apresenta o nome, direção e a designação de cada pino.

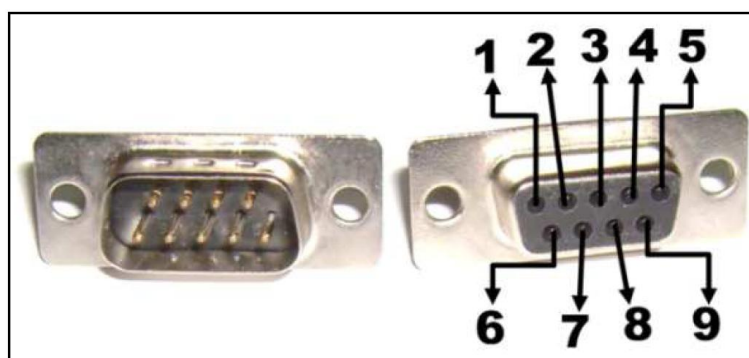


Figura 4.7– Conector DB-9 com sua pinagem. (AUTOR)

Tabela 4.4 – Conector DB9 com direção e designação dos pinos.

Número	Nome	Direção	Designação
1	CD – Carrier Detect	Entrada	Detecção de Portador
2	RXD – Receive Data	Entrada	Recepção de Dados
3	TXD – Transmit Data	Saída	Transmissão de Dados
4	DTR – Data Terminal Ready	Saída	Terminal Pronto
5	GND – Signal Ground	-	Massa Lógica
6	DSR – Data Set Ready	Entrada	Dados Prontos
7	RTS – Request to Send	Saída	Pedido de Emissão
8	CTS – Clear to Send	Entrada	Empréstimo a Emitir
9	RI – Ring Indicator	Entrada	Indicador de Campanha Elétrica

FONTE: ZANCO, 2010.

Para a comunicação do microcontrolador PIC com o módulo SIM900D foi utilizado o CI MAX232. O seu uso foi necessário, pois é um conversor de nível TTL para RS-232. Com isso, apenas três pinos dos conectores DB9 foram utilizados: o pino 2 (RXD), que recebe os dados do módulo, o pino 3 (TXD) que transmite os dados para o módulo e o pino 5 (GND) que é o terra.

### 4.3.1 Circuito Integrado MAX232

Devido ao microcontrolador PIC16F877A e o módulo GSM SIM900D terem níveis de tensão diferenciados, se fez necessário a utilização do CI MAX232, que é um circuito integrado que converte os níveis de tensão correspondentes à lógica TTL no padrão RS-232 e vice-versa. (ZANCO, 2010)

A Figura 4.8 ilustra o CI MAX232 utilizado neste projeto e a Tabela 4.5 apresenta os níveis de tensão correspondentes aos níveis lógicos zero (0) e um (1) que trafegam no padrão RS-232 e os níveis lógicos TTL correspondentes fornecidos pelo microcontrolador PIC16F877A para um tensão de 5 V.



Figura 4.8 – MAX232 utilizado no projeto (AUTOR)

Tabela 4.5 – Níveis de tensão – lógica TTL e padrão RS-232

Nível Lógico	PIC16F877A	Padrão RS-232
0	$\leq 0,6$	+5V a +15V
1	$\geq 5V - 0,6V$	-5V a -15V

FONTE: ZANCO, 2010.

### 4.3.2 Pinagem do CI MAX232

Na Figura 4.9 é mostrado a pinagem do CI MAX232, o interfaceamento dele é fácil, necessitando apenas de quatro capacitores de  $1\mu\text{F}$  e um capacitor de  $10\mu\text{F}$  conectados externamente a alguns dos seus pinos.

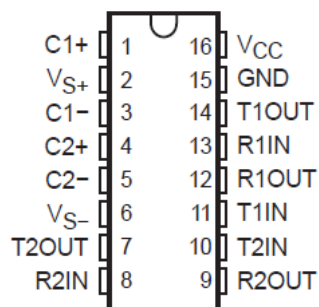


Figura 4.9 – Pinagem do MAX 232  
(FONTE: Texas Instruments, 2004)

A pinagem escolhida para o circuito integrado MAX232 no projeto, como mostrada na Figura 4.10 foi:

- Pino 1 e 3: capacitor de  $1\mu\text{F}$ ;
- Pino 2 e Fonte de alimentação: capacitor de  $1\mu\text{F}$ ;
- Pino 4 e 5: capacitor de  $1\mu\text{F}$ ;
- Pino 6 e Terra: capacitor de  $1\mu\text{F}$ ;
- Pinos 7, 8, 9 e 10: não são utilizados;
- Pino 11: ligação com PIC16F877A no pino 25 (TX);
- Pino 12: ligação com PIC16F877A no pino 26 (RX);
- Pino 13: ligação com a porta serial (macho) no pino 2 (TXD);
- Pino 14: ligação com a porta serial (macho) no pino 3 (RXD);
- Pino 15: aterrado;
- Pino 16: Fonte de alimentação e capacitor de  $10\mu\text{F}$  aterrado.

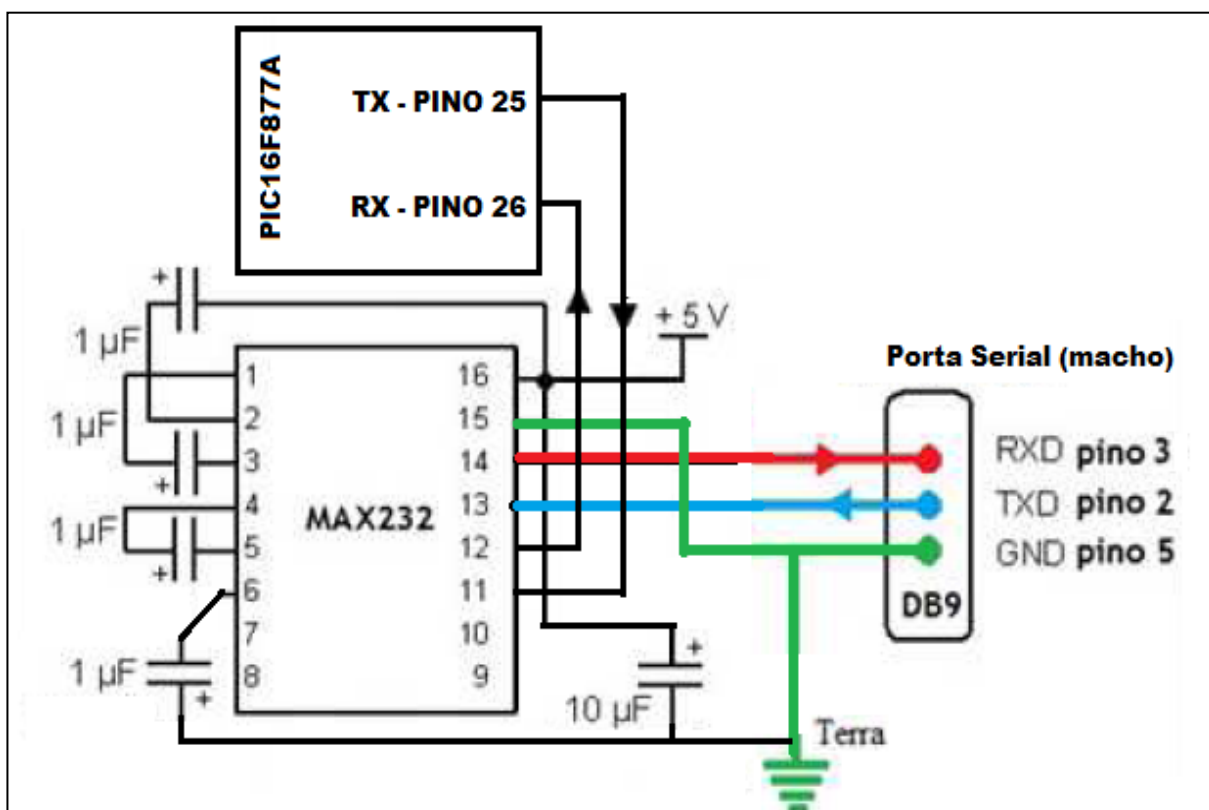


Figura 4.10– Pinagem do MAX232 utilizada no projeto. (AUTOR)

#### 4.4 Teclado Numérico

Foi utilizado um teclado numérico matricial de 12 teclas. Seu funcionamento é de fácil compreensão, através de sete pinos, sendo três pinos referentes às colunas e quatro pinos referentes às linhas, ao pressionar uma tecla o circuito se fecha, fazendo com que a tensão circule de um pino referente à coluna para um pino referente à linha.

Na Figura 4.11 é mostrado o teclado número utilizado no projeto e o diagrama do circuito e a Tabela 4.6 apresenta a pinagem do teclado.

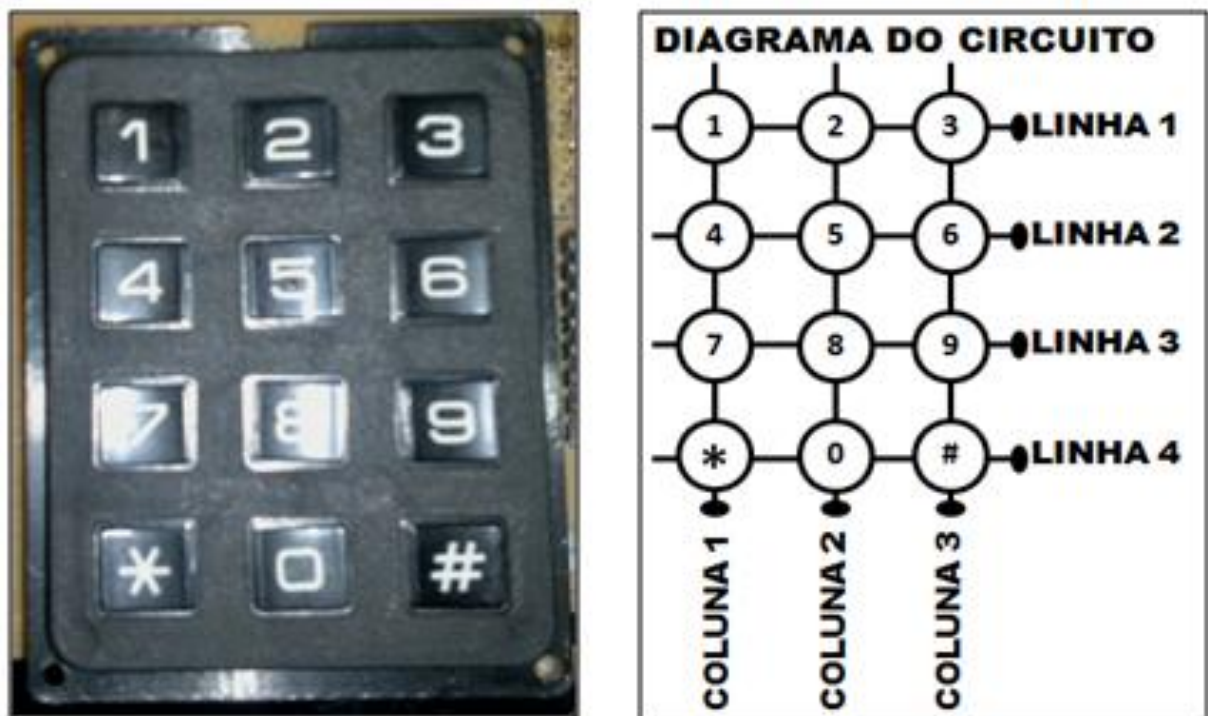


Figura 4.11 – Teclado Numérico e Diagrama do Circuito (AUTOR)

Tabela 4.6 – Pinagem do teclado numérico

Número do Pino	Descrição
1	Linha 1
2	Linha 2
3	Linha 3
4	Linha 4
5	Coluna 1
6	Coluna 2
7	Coluna 3

FONTE: AUTOR.

Para o melhor funcionamento do teclado numérico e a diminuição de ruídos que comprometeriam o normal funcionamento do teclado numérico, como mostrada na Figura 4.12 foi:

- Pinos 1, 2, 3 e 4 – Ligados em paralelo a resistores de  $10\text{k}\Omega$  aterrados cada um e aos pinos 15, 16, 17 e 18 do microcontrolador PIC, respectivamente.
- Pinos 5, 6 e 7 – Ligados em série a resistores de  $470\Omega$  cada um e aos pinos 19, 20 e 21 do microcontrolador PIC, respectivamente.

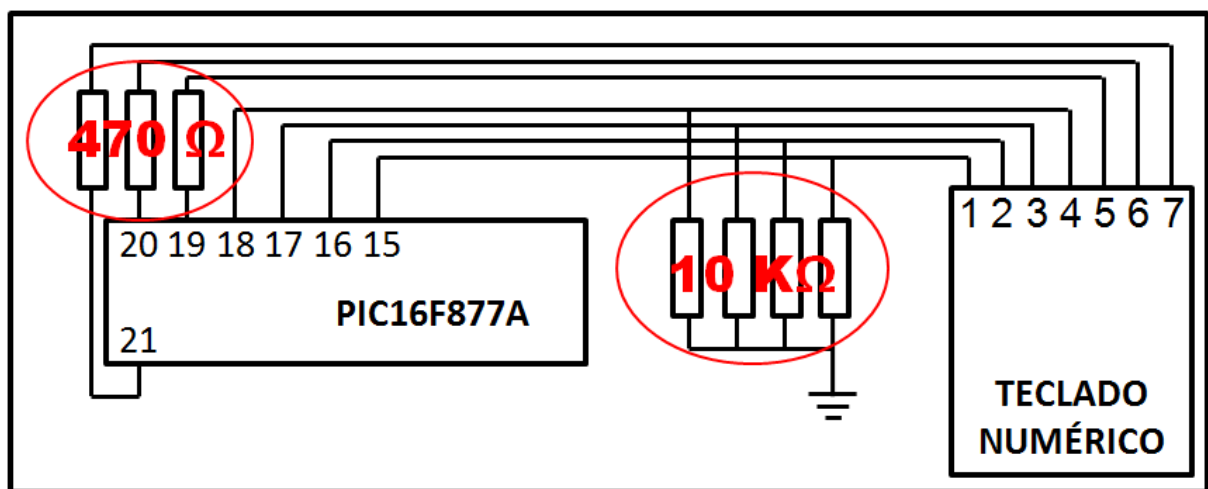


Figura 4.12 – Interligação do teclado numérico utilizado no projeto. (AUTOR)

#### 4.5 Módulo LCD

Existe uma variedade de módulos LCD disponíveis no mercado, para este projeto foi escolhido o LCD 16X2, ou seja, o módulo possui duas linhas, cada linha sendo possível a escrita de dezesseis caracteres. A Figura 4.13 ilustra o módulo LCD utilizado no projeto.



Figura 4.13– Módulo LCD utilizado no projeto (AUTOR)

Como se pode observar o módulo possui 16 pinos, a Tabela 4.7 apresenta a descrição dos pinos do módulo LCD e o interfaceamento no projeto.

**Tabela 4.7 – Pinos do módulo LCD e interfaceamento no projeto**

Pino	Símbolo	Função	Interfaceamento no projeto
16	K	Cátodo	NÃO UTILIZADO
15	A	Anodo	NÃO UTILIZADO
1	VSS	Terra	Aterrado
2	VDD	5 V	Fonte de Alimentação
3	V <sub>O</sub>	Ajuste de contraste	Potenciômetro
4	R/S	Seleção de registro	Pino 08 do PIC16F877A
5	R/W	Leitura / Escrita	Aterrado
6	E	Inicia ciclo Leitura / Escrita	Pino 09 do PIC16F877A
7	DB0	Dado	NÃO UTILIZADO
8	DB1	Dado	NÃO UTILIZADO
9	DB2	Dado	NÃO UTILIZADO
10	DB3	Dado	NÃO UTILIZADO
11	DB4	Dado	Pino 27 do PIC16F877A
12	DB5	Dado	Pino 28 do PIC16F877A
13	DB6	Dado	Pino 29 do PIC16F877A
14	DB7	Dado	Pino 30 do PIC16F877A

FONTE: ZANCO, 2010.

Foi utilizado apenas quatro bits de dados, as quatro linhas mais significativas (DB7:DB4), com isso foram necessários apenas seis pinos do microcontrolador para que a comunicação com o LCD fosse viabilizada. Com isso, a transferência de um byte para o LCD é feita em duas partes: primeiro é enviado o *nibble*<sup>7</sup> alto do byte (DB7:DB4), depois é enviado o *nibble* baixo (DB3:DB0). (ZANCO, 2010, p. 99).

#### 4.6 Sensor LDR (*Light Dependent Resistor*)

O sensor LDR (Resistor Dependente de Luz), nada mais é do que um resistor que sua resistência depende da luz que o atinge, com isso ao ser colocado no escuro sua resistência passa para ordem de milhões de ohms, já quando recebe iluminação direta, luz forte ou luz direta do Sol sua resistência é diminuída para algumas centenas de milhares de ohms. (ALBUQUERQUE e THOMAZINI, 2009).

<sup>7</sup> Nibble – é a sucessão de quatro cifras binárias (bits).

Os LDRs não são componentes polarizados, o que significa que a corrente pode circular nos dois sentidos. As variações da resistência com a luz são iguais em qualquer sentido. (ALBUQUERQUE e THOMAZINI, 2009).

Na Figura 4.14 é mostrado o sensor LDR utilizado no projeto e a variação da resistência do LDR em função luminosidade.

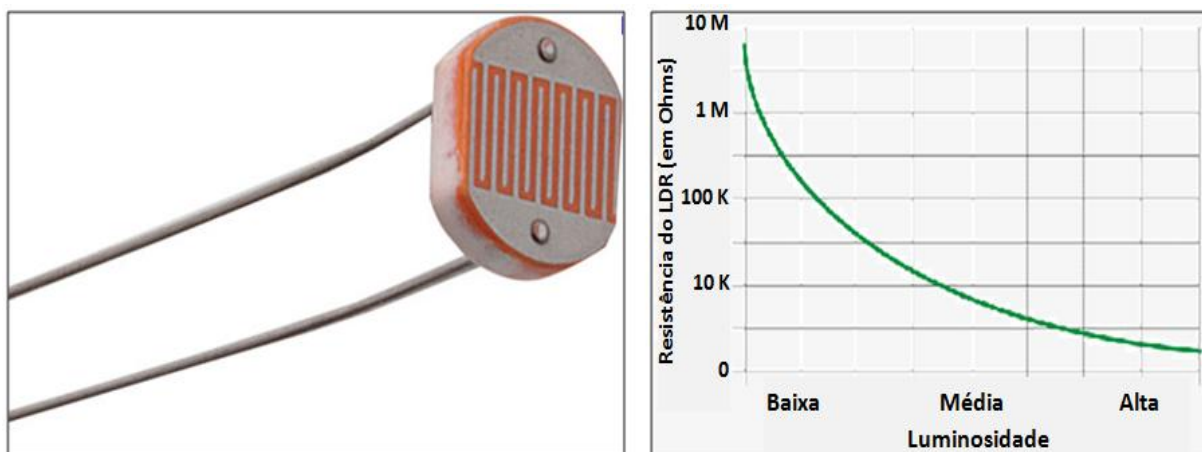


Figura 4.14 – Sensor LDR utilizado no projeto e variação de resistência com a luz. (AUTOR)

#### 4.7 Relé

Os relés são dispositivos eletromecânicos capazes de controlar circuitos de grandes correntes a partir de pequenas correntes ou tensões. Seu funcionamento é bem simples, quando uma corrente circula pela bobina, esta cria um campo magnético que atrai o contato fechando ou abrindo circuito. Ao cessar a corrente da bobina o campo magnético também cessa e o contato volta a sua posição original.

Na Figura 4.15 são mostrados os relés utilizados no projeto.

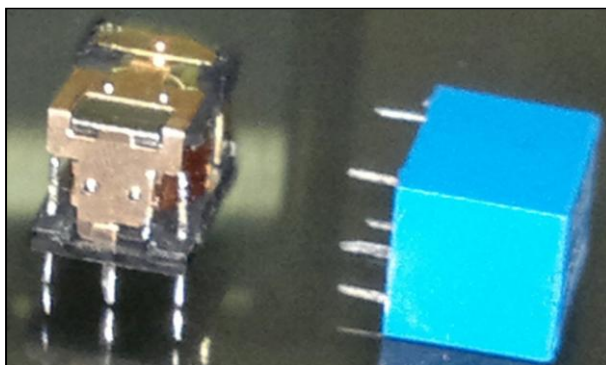


Figura 4.15 – Relés utilizados no projeto (AUTOR)



## CAPÍTULO 5 – IMPLEMENTAÇÃO

Neste capítulo são apresentados tópicos fundamentais para compreensão geral da implementação do projeto, foram definidas as etapas necessárias para isso, que são:

- Modelagem do sistema;
- Elaboração dos circuitos;
- Elaboração do código fonte para o microcontrolador PIC;
- Montagem dos circuitos nas placas;
- Montagem do protótipo;

Na Figura 5.1 é mostrado o protótipo na sua fase final.

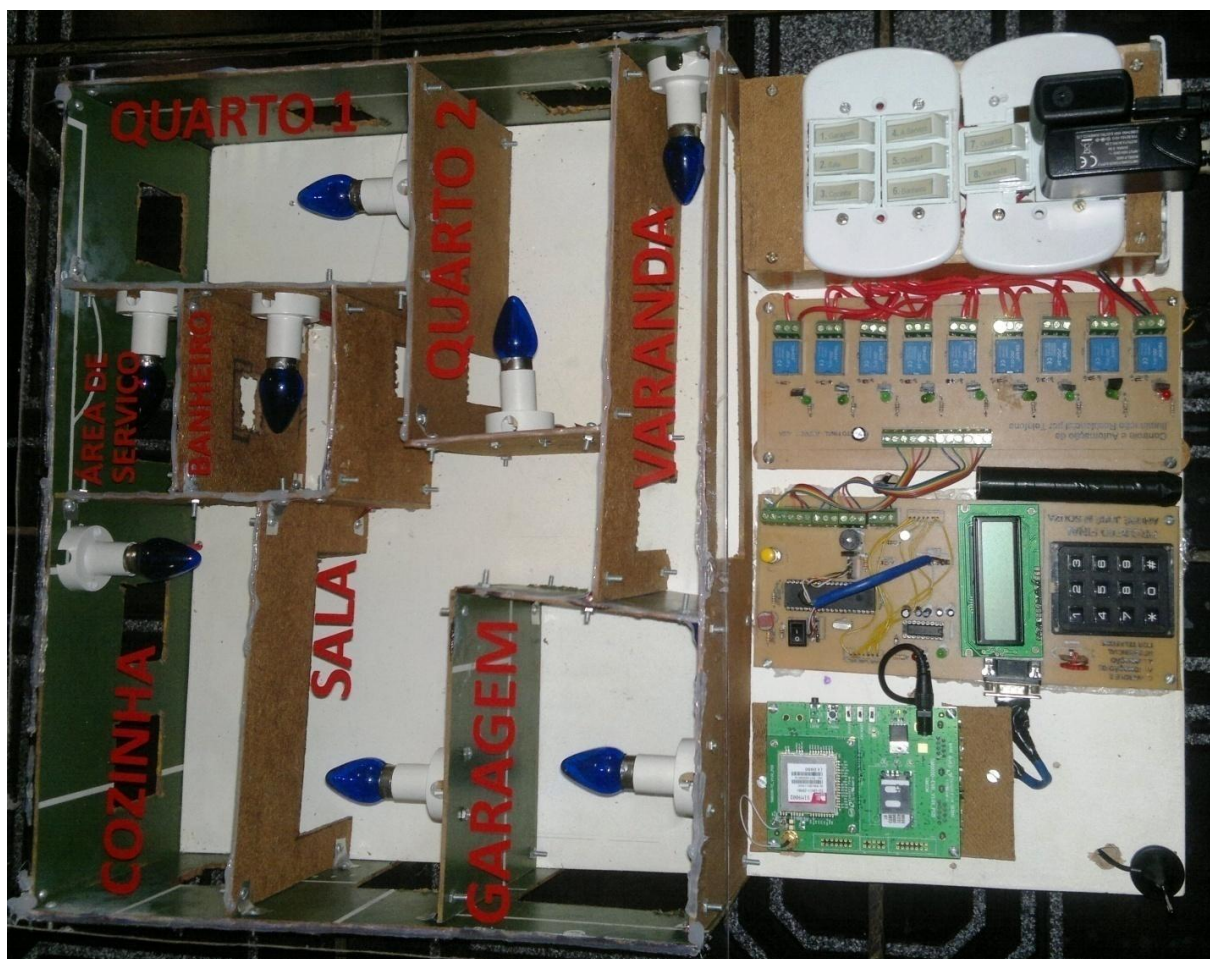


Figura 5.1 – Protótipo na Fase Final (AUTOR)



## 5.1 Modelagem do Sistema

A primeira etapa na implementação deste projeto foi à realização da modelagem do sistema, ou seja, como cada dispositivo estaria disposto e como seria feita a demonstração em um sistema real, que utilize a corrente elétrica captada pelas linhas de transmissão local (220 Volts de Corrente Alternada).

Para isso foi pensado a elaboração da maquete de uma casa, na qual temos a representação de oito cômodos, tendo em cada cômodo uma lâmpada de 220 Volts, na mesma estrutura temos a fixação dos interruptores, tomadas e demais dispositivos utilizados.

Para montagem do protótipo foi obtido pelo autor um tablado de madeira de 45 centímetros por 60 centímetros, no qual a casa, interruptores e tomadas preencheram 70% do espaço, deixando o espaço restante para a fixação dos demais dispositivos que seriam o módulo GSM, a placa principal, e a placa de relés.

A Figura 5.2 ilustra a disposição dos componentes e a planta baixa da casa com sua respectiva escala.

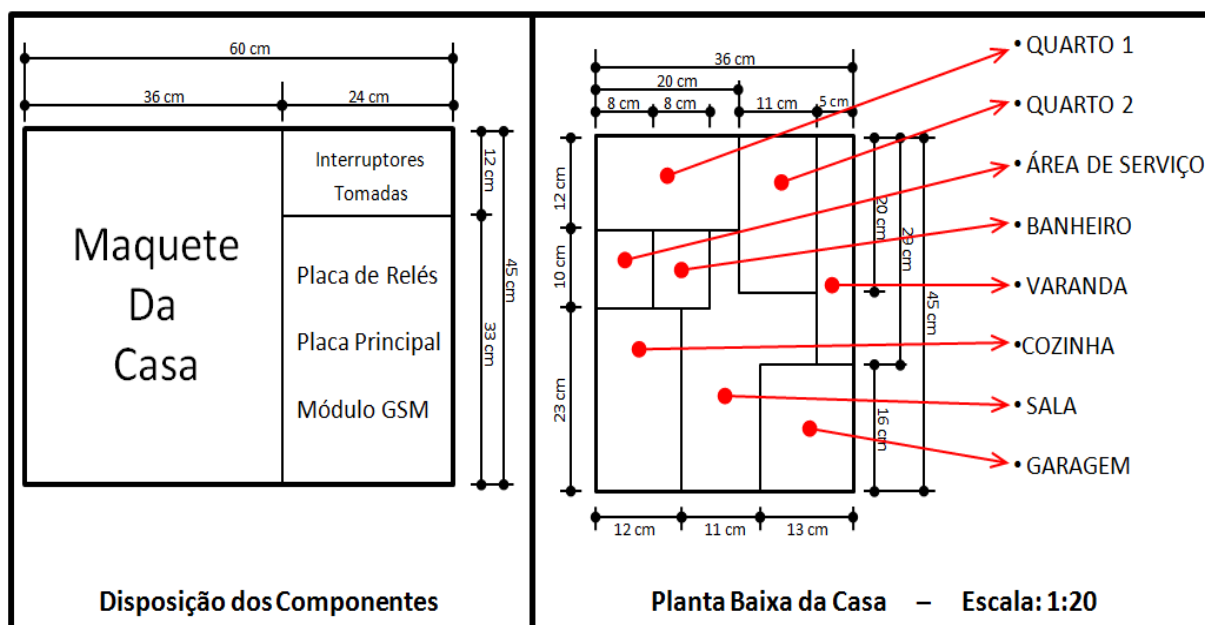


Figura 5.2 – Disposição dos Componentes e Planta Baixa da Casa (AUTOR)

### 5.1.1 – Fluxograma Geral do Sistema

Após definição da disposição dos componentes, foi elaborado o fluxograma geral do sistema, conforme mostra a Figura 5.3. Esse fluxograma foi fundamental na elaboração do código fonte e na compra de todos os componentes que foram utilizados na elaboração dos circuitos e na montagem final do protótipo.

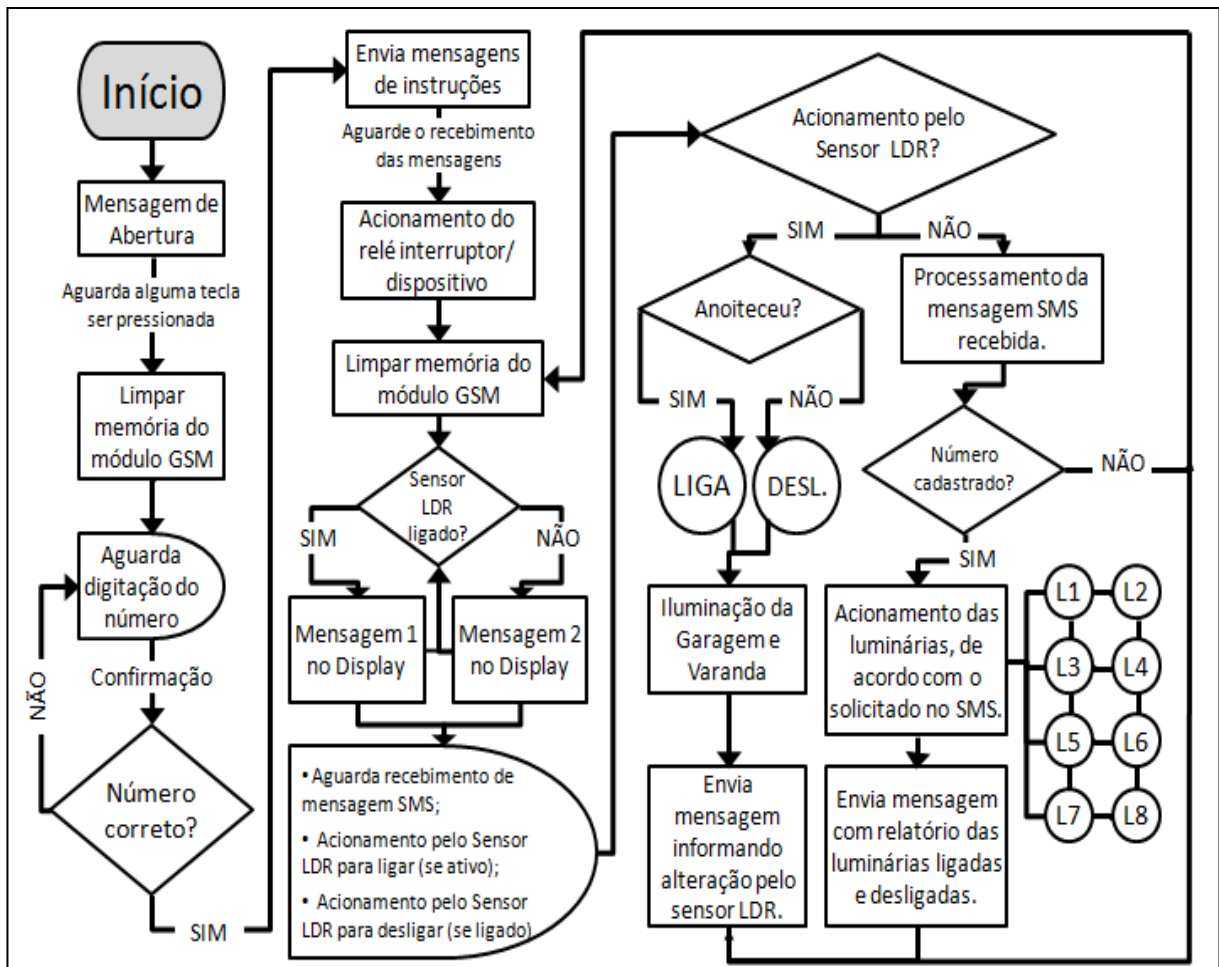


Figura 5.3 – Fluxograma Geral do Sistema (AUTOR)

## 5.2 Elaboração dos Circuitos

Com o fluxograma feito o próximo passo foi a elaboração dos circuitos, para isso foi utilizado, primeiramente, o programa Proteus ISIS 7 Profissional. No qual neste primeiro momento foi possível definir quais seriam os componentes utilizados, e as quais pinos do microcontrolador PIC eles seriam interligados. A Figura 5.4 ilustra o circuito da placa principal que foi utilizada para os testes iniciais.

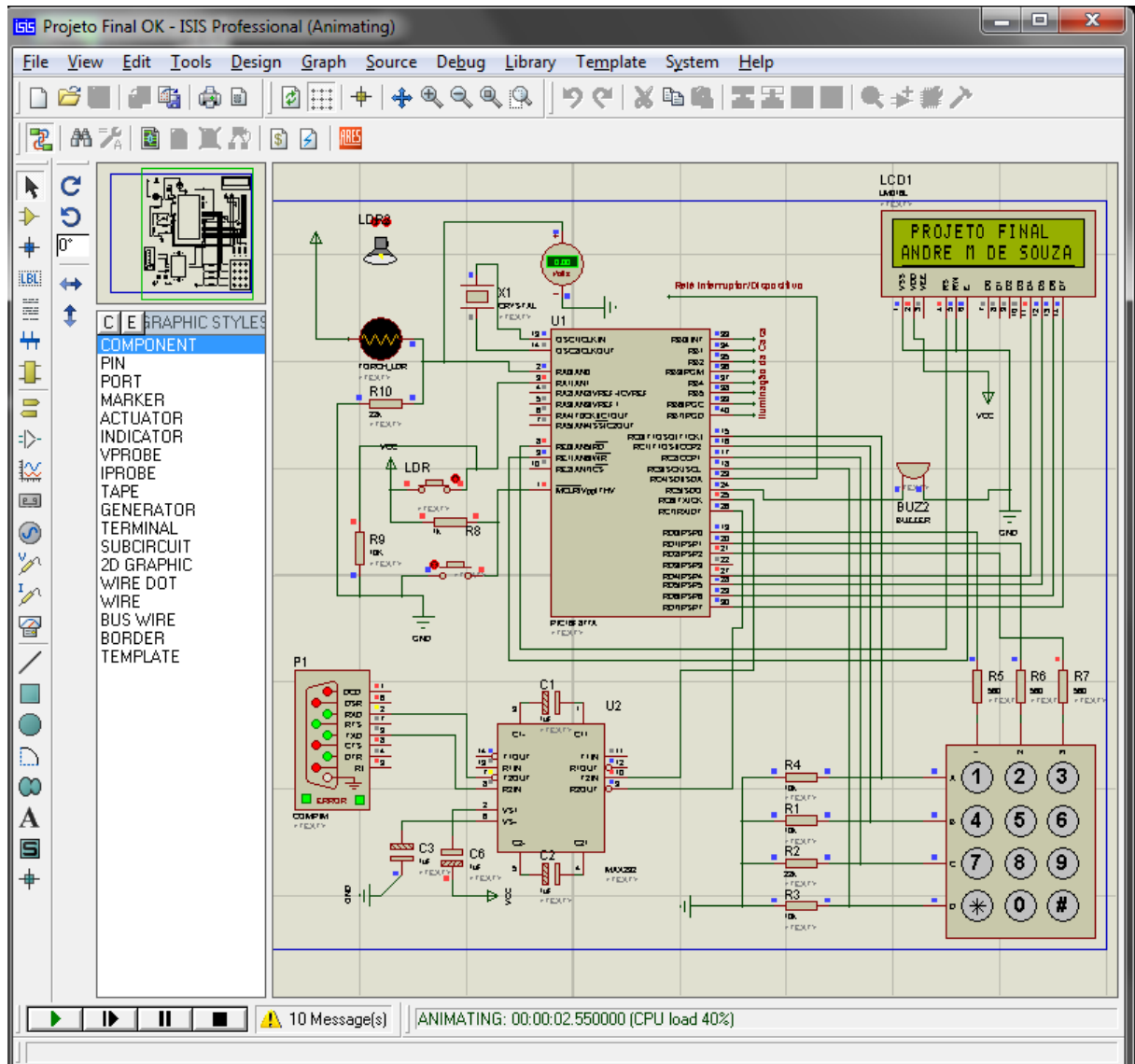


Figura 5.4 – Circuito da Placa Principal feita no Proteus ISIS (AUTOR)

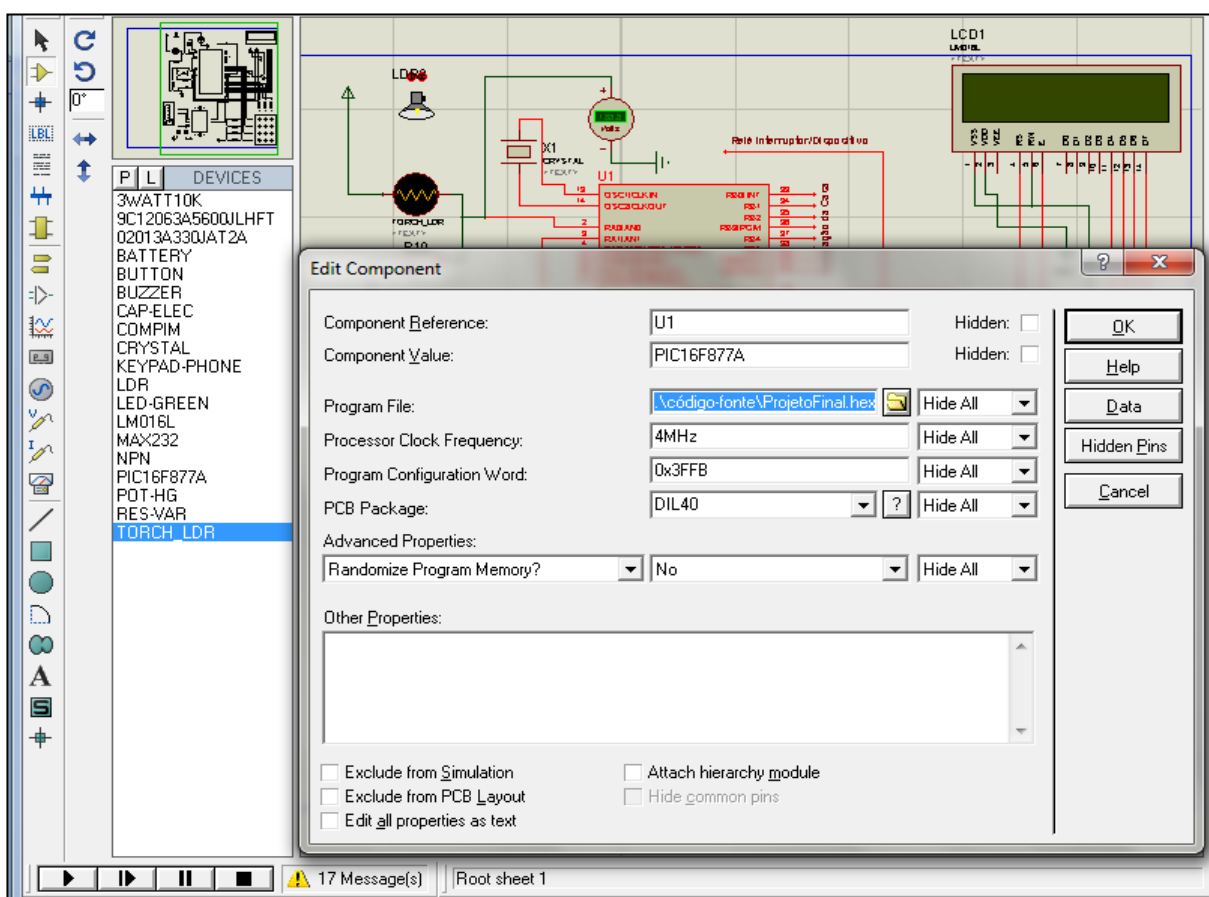
### 5.2.1 Proteus ISIS 7 Professional<sup>1</sup>

O Proteus ISIS é um simulador de circuitos eletrônicos, no qual é possível ver o seu código interagir com os componentes em tempo real, tem em seu banco de dados uma série de componentes, dos mais simples, como resistores, capacitores, aos mais complexos com circuitos integrados como o MAX232 e microcontroladores.

<sup>1</sup> Proteus ISIS 7 Professional – é um dos programas de um suíte desenvolvido pela empresa inglesa *Labcenter Electronics*, fundada em 1988 pelo presidente John Jameson. (página na Web: <http://www.labcenter.com>)

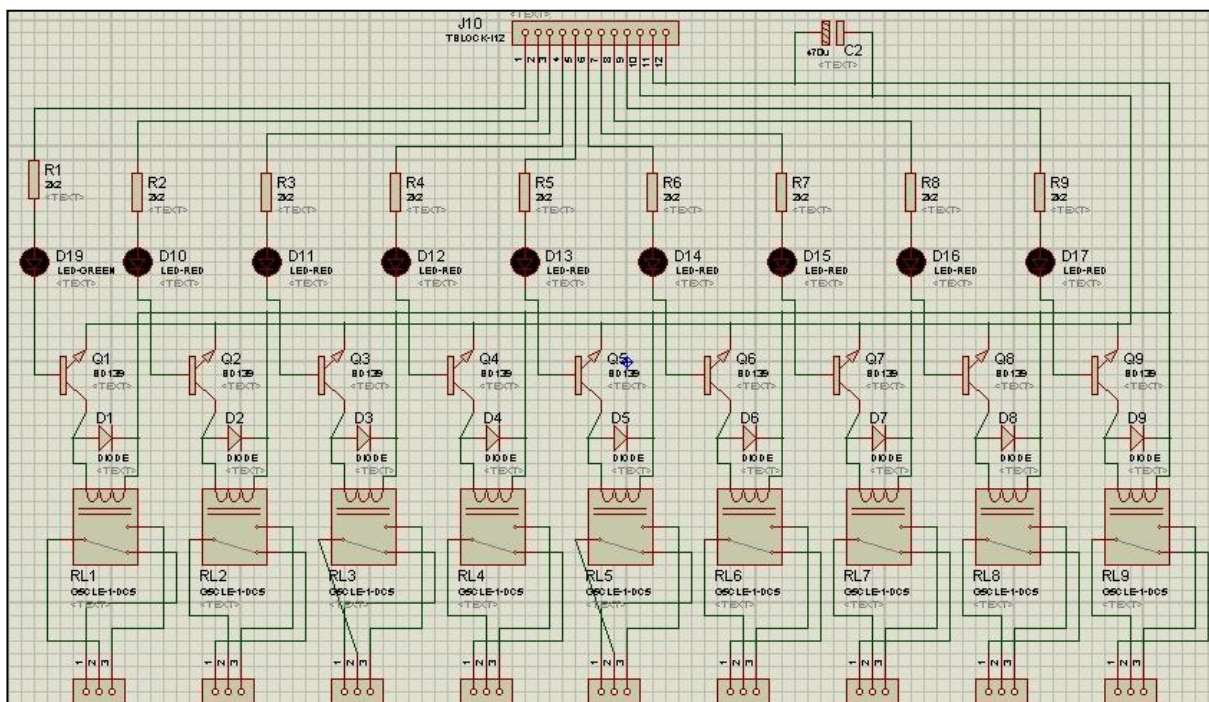
Na simulação pelo Proteus ISIS deste projeto, os componentes de maior relevância utilizados, foram: o módulo LCD 16X2, teclado telefônico matricial, circuito integrado MAX232, para sensor LDR foram testados os dois tipos presentes e o microcontrolador PIC16F877A.

Para realizar a simulação uma das funcionalidades do Proteus ISIS é a ‘gravação’ do código fonte no microcontrolador PIC conforme mostrado na Figura 5.5. Para isso é necessário que o código já esteja compilado para o formato hexadecimal, compilação essa realizada pelo CCS C *Compiler*, conforme mostrado na seção – CCS C *Compiler* – do capítulo anterior.



**Figura 5.5 – Inserção do código fonte (AUTOR)**

Para confecção da placa de relés os componentes utilizados foram: Relé de 5vcc, diodo simples, transistor NPN, resistor de 1kΩ, LED, terminal de três saídas, como foi necessário nove relés, foram necessários nove componentes de cada um listado a cima, além de um capacitor de 470μF para desacoplamento e um terminal de doze saídas. Na Figura 5.6 é mostrado o circuito da placa de relés feito no Proteus ISIS.



**Figura 5.6 – Circuito da Placa de Relés feita no Proteus ISIS (AUTOR)**

Mesmo o Proteus ISIS sendo um programa que se mostrou de grande importância na elaboração dos circuitos, algumas etapas referentes à simulação, como comunicação serial, sensibilidade do sensor LDR, funcionamento do relé em conjunto com transistor deixaram a desejar. Com isso foi necessário a elaboração dos circuitos em um sistema físico para realização de novos testes.

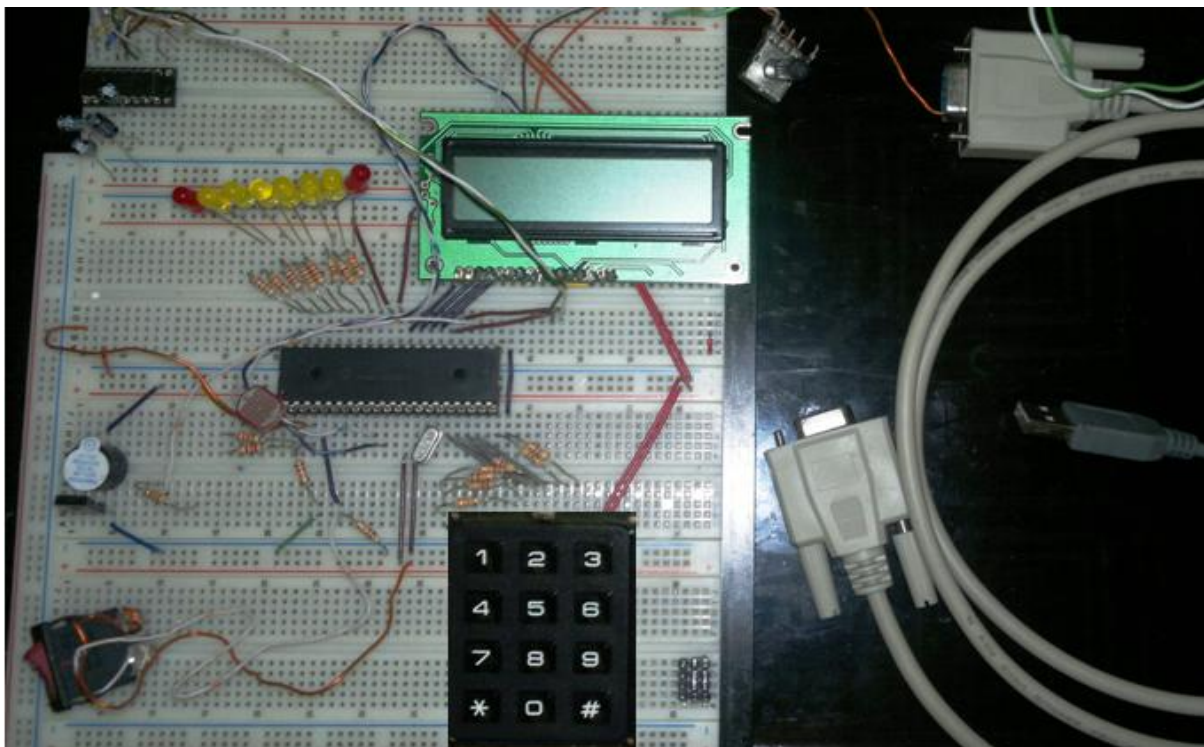
### 5.2.2 Elaboração dos Circuitos na Matriz de Contato<sup>2</sup>

Em uma segunda etapa, após compra dos componentes que seriam utilizados, os circuitos antes feitos no Proteus ISIS foram duplicados em uma matriz de contato, para que assim os testes antes feitos no programa pudessem ser refeitos com a utilização dos componentes em um meio físico.

Na Figura 5.7 é mostrada a interligação dos componentes da placa principal na matriz de contato, para fins de testes nas saídas dos pinos que controlam a iluminação dos cômodos, foram colocados LEDs e resistores.

<sup>2</sup> Matriz de Contato (ou *Protoboard*) – é uma placa com furos e conexões condutoras para montagem de circuitos eletrônicos experimentais, tendo como vantagem a inserção de componentes sem a necessidade de soldagem.



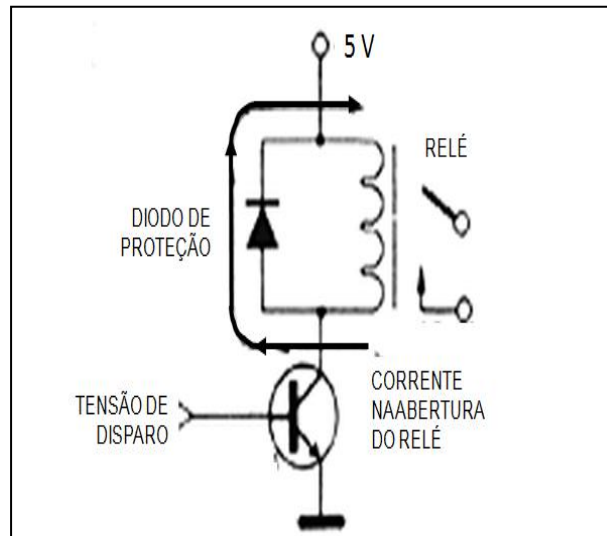


**Figura 5.7 – Circuito da Placa Principal feita na Matriz de Contato (AUTOR)**

Outra alteração em relação ao circuito feito no Proteus ISIS foi à utilização de transistor NPN para acionamento do sinal sonoro (*buzzer*), a utilização de um capacitor de desacoplamento no CI MAX232, para desvio de possíveis ruídos da fonte. Como fonte de alimentação principal nesta fase de testes na matriz de contato foi utilizada a porta USB do computador pessoal do autor, na saída do pino 01 do microcontrolador PIC foi utilizado um fio que quando aterrado reiniciava o sistema.

No caso do circuito da placa de relés, de acordo com Braga (2010?), no momento em que o relé é desenergizado, as linhas de força do campo magnético da bobina, começam a se contrair, havendo então a indução de uma tensão. Esta tensão tem polaridade oposta a do sistema e pode atingir valores muito altos.

Segundo Braga (2010?), para proteger os componentes de uma possível queima, a técnica mais empregada é o uso de um diodo. O que ocorre é que o diodo está polarizado inversamente em relação à tensão de disparo do relé. Assim, quando ocorre a indução de uma alta tensão nos extremos da bobina no momento da interrupção da corrente, o diodo polarizado no sentido direto passa a ter uma baixa resistência absorvendo assim a energia que poderia afetar os componentes de disparo, conforme mostrado na Figura 5.8.



**Figura 5.8 – Relé e diodo de proteção.**  
(FONTE: BRAGA, 2010?)

### 5.3 Elaboração do Código Fonte para o Microcontrolador PIC

Com as etapas, modelagem do sistema e elaboração dos circuitos, concluídas, a próxima etapa foi à elaboração do código fonte e gravação no microcontrolador. Para escrita das linhas de comando e compilação do código foi utilizado o programa CCS C *Compiler* e para gravação do código a ferramenta utilizada foi o kit de gravação PICkit 2 da Robótica Simples, que é composto por dispositivo e programa.

#### 5.3.1 Escrita do Código Fonte

Todo o código fonte foi escrito utilizando a linguagem C, devido principalmente às facilidades que tanto a linguagem de programação C como o próprio programa utilizado no desenvolvimento do código trazem.

“... a utilização de uma linguagem de alto nível como C permite que o programador preocupe-se mais com a programação da aplicação em si, já que o compilador assume para si tarefas como o controle e localização das variáveis, verificação de bancos de memória, etc.” (PEREIRA, 2007, p.18).

Com o fluxograma geral do sistema que foi apresentado na Figura 5.3 e com as interligações pretendidas na elaboração do circuito, o primeiro passo é definir as diretivas de pré-compilação, definições estas que são de extrema importância para a correta compilação do código.

```

1  /*=====
2  DIRETIVAS DE PRÉ-COMPILAÇÃO
3  =====*/
4  #include <16f877a.h>           // Diretivas do PIC16F877A
5  #device adc=10                // Configuração do ADC
6  #use delay(clock=4000000)     // Informa a velocidade do clock
7  #fuses xt, nowdt, put         // Fusíveis
8  #fuses nolvp, nowrt, nocpd    // Fusíveis
9  #fuses nobrownout, noprotect  // Fusíveis
10 #use rs232 (baud=9600,xmit=pin_C6,rcv=pin_C7,restart_wdt,errors,stream=modem)
11                               // Ativa suporte comunicação serial RS232
12 #include <mod_lcd.c>           // Diretivas do display LCD
13 #include <string.h>           // Requisito para Comparar duas strings.
14 #define sensib                300 // Sensibilidade utilizada para o sensor LDR
15 #define LDR                    PIN_A1 // Liga e desliga o sensor LDR
16 #define LUM1                   PIN_B0 // Luminária 1 - Garagem
17 #define LUM2                   PIN_B1 // Luminária 2 - Sala
18 #define LUM3                   PIN_B2 // Luminária 3 - Cozinha
19 #define LUM4                   PIN_B3 // Luminária 4 - Área de Serviço
20 #define LUM5                   PIN_B4 // Luminária 5 - Quarto 1
21 #define LUM6                   PIN_B5 // Luminária 6 - Banheiro
22 #define LUM7                   PIN_B6 // Luminária 7 - Quarto 2
23 #define LUM8                   PIN_B7 // Luminária 8 - Varanda
24 #define LIN1                   PIN_C0 // Teclado Matricial - pino 1 (linha 1)
25 #define LIN2                   PIN_C1 // Teclado Matricial - pino 2 (linha 2)
26 #define LIN3                   PIN_C2 // Teclado Matricial - pino 3 (linha 3)
27 #define LIN4                   PIN_C3 // Teclado Matricial - pino 4 (linha 4)
28 #define LUM_todas              PIN_C4 // Luminárias (Interruptor/Dispositivo)
29 #define BUZZER                 PIN_C5 // Alarme para uso geral
30 #define COL1                   PIN_D0 // Teclado Matricial - pino 5 (coluna 1)
31 #define COL2                   PIN_D1 // Teclado Matricial - pino 6 (coluna 2)
32 #define COL3                   PIN_D2 // Teclado Matricial - pino 7 (coluna 3)
33

```

Figura 5.9 – Código fonte, diretivas de pré-compilação. (AUTOR)

Na Figura 5.9 são mostradas as diretivas utilizadas no código fonte e a frente de cada linha de comando, após duas barras conjuntas um prevê comentário da diretiva utilizada. A tabela 5.1 apresenta a descrição de cada comando utilizado.

Tabela 5.1 – Descrição de diretivas do código fonte.

Comando	Descrição
#include	Insere um arquivo texto externo a partir da posição atual.
#use delay	Informa ao compilador a velocidade de <i>clock</i> do sistema de destino.
#use rs232	Ativa o suporte à comunicação serial. Especificam: velocidade, pinos TX e RX.
#fuses	Programa as opções da palavra de configuração ( <i>configuration word</i> ) do PIC.
#device	Define o nome do processador utilizado.
#define	Substitui o identificador pelo texto especificado imediatamente depois dele.

FONTE: PEREIRA, 2007.

Na elaboração do código fonte, todas as funções básicas necessárias na programação, foram acrescentadas após a função principal. Com isso, foi necessário realizar a chamada dessas funções, no início do programa, conforme mostrado na Figura 5.10, que ilustra também as variáveis utilizadas na programação.



```

44  // CHAMADA INICIAL DAS FUNÇÕES BÁSICAS
45  //=====*/
46  void Porta_Serial(void);           // Interrupção, recebendo comunicação serial
47  void limpa_GSM(void);             // Limpa mensagens SMS armazenadas no chip
48  void SMS_instrucoes1(void);       // Mensagem SMS, primeiras instruções
49  void SMS_instrucoes2(void);       // Mensagem SMS, estado das luminárias
50  void SMS_instrucoes3(void);       // Mensagem SMS, alteração pelo LDR, ligado
51  void SMS_instrucoes4(void);       // Mensagem SMS, alteração pelo LDR, desligado
52  char varre_tecclas(void);         // Varre o tecla numérico
53  void SoaBuzzer(int w);            // Toca o buzzer.
54  int FuncaoDeParada(void);         // Varre teclas * e #, se *=0, se #=1
55
56  /*=====
57  // VARIÁVEIS
58  //=====*/
59  long int sensorLDR;
60  char telefone[10], telefone2[10], luminaria[9], buffer1[74], buffer2[80];
61  int i, j, tecla, indice1=0, indice2=0;
62  boolean LDR_controle=0, LDR_controle1=0, testar=0;
63  boolean iLUM1=0, iLUM2=0, iLUM3=0, iLUM4=0;
64  boolean iLUM5=0, iLUM6=0, iLUM7=0, iLUM8=0;
65  boolean LDR1=0, LDR0=0, tem_dado=0, SMS=0;

```

Figura 5.10 – Código fonte, chamada de funções e variáveis (AUTOR)

Na Figura 5.11 é mostrada a principal função básica utilizada na programação do código. Que define o tratamento da interrupção quando houver recepção de dados na comunicação serial.

```

461  #int_rda // Interrupção de recepção de dados
462  void Porta_Serial(void){
463      char dado;
464      dado=getc();
465      if(indice1<74){
466          buffer1[indice1]=dado;
467          indice1++;}
468      else{
469          buffer2[indice2]=dado;
470          indice2++;}
471      if(dado==10){
472          tem_dado=1;}

```

Figura 5.11 – Código fonte, função para uso de interrupção. (AUTOR)

Para utilização desta função, foi necessário habilitar a interrupção, com a utilização das linhas de comando: (PEREIRA, 2007).

```

enable_interrupts(int_rda);
enable_interrupts(global);

```

Basicamente, ao receber um dado na comunicação serial, o programa é desviado para a interrupção. Na programação utilizada pelo autor, cada caractere é adicionado na *string buffer1*, após chegar ao seu tamanho total, passa a ser adicionado na *string buffer2*. Quando este caractere é do tipo *Line Feed* (Próxima Linha), a valor da variável *tem\_dado* é mudado.

Para elaboração das funções de envio de mensagem e a leitura de mensagens SMS recebidas, o programa *HyperTerminal* se fez necessária para a realização de testes dos comandos AT e na visualização dos dados recebidos que seriam utilizados na programação do código fonte. Com esses testes foi possível verificar que ao receber uma mensagem SMS, o módulo GSM apresenta a seguinte mensagem:

+CMTI: "SM", 1

Ao realizar a leitura desta mensagem recebida, pelo comando: *AT+CMGR=1*, o módulo GSM apresenta a seguinte mensagem:

+CMGR: "REC UNREAD", "+556198765432", "", "10/10/11,12:12:12+00"  
A MENSAGEM ENVIADA

Com isso é possível, além da leitura da mensagem, verificar o número do remetente da mensagem, dado este que ao longo da programação foi utilizado na comparação com o número cadastrado no sistema, para aferição de que a mensagem recebida foi realmente do cliente cadastrado.

Para envio de mensagens SMS o autor seguiu as instruções do manual de comandos AT do módulo GSM SIM900D, disponível no site da distribuidora brasileira – ME Componentes (página na Web: <http://www.mecomp.com.br>)

```

318 void SMS_instrucoes1(void) { // 2 mensagens SMS (instruções e SMS padrão)
319     fprintf(modem, "AT\r");
320     delay_ms(300);
321     fprintf(modem, "AT+CMGF=1\r");
322     delay_ms(300);
323     fprintf(modem, "AT+CSCS=\"GSM\"\r");
324     delay_ms(300);
325     fprintf(modem, "AT+CMGS=\"");
326     fprintf(modem, "%s", telefone);
327     fprintf(modem, "\"\r");
328     delay_ms(300);
329     fprintf(modem, "PARA LIGAR/DESLIGAR A ILUMINAÇÃO, ENVIE A ");
330     fprintf(modem, "PROXIMO MENSAGEM A 99788538, SUBSTITUINDO O 'X' POR ");
331     fprintf(modem, "'1' PARA LIGADO E '0' PARA DESLIGADO.");
332     delay_ms(300);
333     putc(26); putc(26); putc(26);
334     delay_ms(4500);
335     fprintf(modem, "AT\r");
336     delay_ms(300);
337     fprintf(modem, "AT+CMGF=1\r");
338     delay_ms(300);
339     fprintf(modem, "AT+CSCS=\"GSM\"\r");
340     delay_ms(300);
341     fprintf(modem, "AT+CMGS=\"");
342     fprintf(modem, "%s", telefone);
343     fprintf(modem, "\"\r");
344     delay_ms(300);
345     fprintf(modem, "X. GARAGEM\rX. SALA\rX. COZINHA\rX. SERVIÇO\r");
346     fprintf(modem, "X. QUARTO1\rX. BANHEIRO\rX. QUARTO2\rX. VARANDA\r");
347     delay_ms(300);
348     putc(26); putc(26); putc(26);

```

Figura 5.12 – Código fonte, função de envio de mensagem SMS (AUTOR)

Na Figura 5.12 é mostrada uma das funções de envio de mensagem SMS utilizadas no código fonte, sendo esta utilizada para enviar duas mensagens SMS, a primeira com instruções de funcionamento do sistema e a segunda com a mensagem SMS padrão que será utilizada pelo cliente.

Outro ponto importante no código fonte é a leitura do pino de entrada do conversor AD interno, para isso é necessário a utilização de algumas diretivas:

```
#device adc=10           // Especifica 10 bits a ser retornado pela função read_adc()
setup_adc(adc_clock_internal); // Configura o conversor AD interno
setup_adc_ports(an0);     // Configura o pino 2 do PIC como conversor AD interno
set_adc_channel(0);       // Seleciona uma canal de entrada para o módulo AD interno
sensorLDR = read_adc();   // Efetua conversão AD, retorna valor de 0 a 1024.
```

A elaboração da função principal seguiu as etapas relatadas no fluxograma geral. Sendo a primeira etapa o cadastro do telefone do cliente, no qual o sistema realiza o envio da mensagem de instruções e da mensagem SMS padrão.

Após envio das mensagens SMS o programa entra em um laço infinito, e logo após entra em outro laço, o laço principal. A Figura 5.13 ilustra este laço principal. De forma resumida, o programa só saíra deste laço, se receber uma mensagem SMS ou o valor da variável de controle do sensor LDR for mudado.

```
137 do{                                     // Loop secundário, aguarda SMS ou Sensor LDR
138     buffer1[0]='\0';                     // Limpa o buffer1
139     buffer2[0]='\0';                     // Limpa o buffer2
140     indice1=indice2=0;                   // Zera os contadores utilizados nos buffers
141     set_adc_channel(0); delay_us(200);
142     sensorLDR = read_adc(); // Efetua conversão AD
143     if (input(LDR) && LDR0==0){ // Mensagem Display, quando sensor LDR ligado
144         lcd_ini();
145         lcd_escreve('\f'); lcd_escreve(" Dispositivo ON ");
146         lcd_escreve('\n'); lcd_escreve(" Sensor LDR ON ");
147         LDR0=1;}
148     if (!input(LDR) && LDR0==1){ // Mensagem Display, quando sensor LDR desligado
149         lcd_ini();
150         lcd_escreve('\f'); lcd_escreve(" Dispositivo ON ");
151         lcd_escreve('\n'); lcd_escreve(" Sensor LDR OFF");
152         LDR0=0;}
153     if (sensorLDR<sensib && LDR_controle==0 && input(LDR)){
154         delay_ms(3000); // Sensor LDR ligado, escureceu.
155         sensorLDR = read_adc(); // Realiza novo teste, e seta a
156         if (sensorLDR<sensib){ // variavel de controle
157             LDR_controle1=1;}}
158     if (sensorLDR>=sensib && LDR_controle==1 && input(LDR)){
159         delay_ms(3000); // Sensor LDR ligado, claridade.
160         sensorLDR = read_adc(); // Realiza novo teste, e seta a
161         if (sensorLDR>=sensib){ // variavel de controle (Tem que
162             LDR_controle1=1;}} // estar ativo devido ao sensor LDR)
163     if (tem_dado==1){ // Caso receba um caractere "Proxima Linha"
164         delay_ms(1000);
165         tem_dado=0;
166         if (buffer1[4]=='T' && buffer1[5]=='I'){ // Verifica se é mensagem SMS
167             SMS=1;}}
168 }while (!(SMS==1) && !(input(LDR) && LDR_controle1==1));
169 //Sairá do WHILE: recebeu uma mensagem SMS, ou variavel de controle LDR setada
```

Figura 5.13 – Código Fonte, laço principal do sistema (AUTOR)

As alterações do estado da iluminação de cada cômodo são realizadas com a leitura pelo sistema da mensagem SMS padrão, por isso é essencial que seja mantida a estrutura original desta mensagem, para que não haja erros na leitura da mensagem. Caso o remetente não seja o mesmo número que realizou o cadastro no sistema, nenhuma alteração é realizada e não ocorre envio de mensagem desnecessária para o proprietário.

### 5.3.2 Compilação e Gravação do Código Fonte

Após escrita do código fonte, o próximo passo é a sua compilação, com a utilização do CCS C Compiler, essa atividade pode ser realizada de duas formas: pressionando a tecla F9 ou através do ícone “Compile” na aba Compile, conforme é mostrado na Figura 5.14.

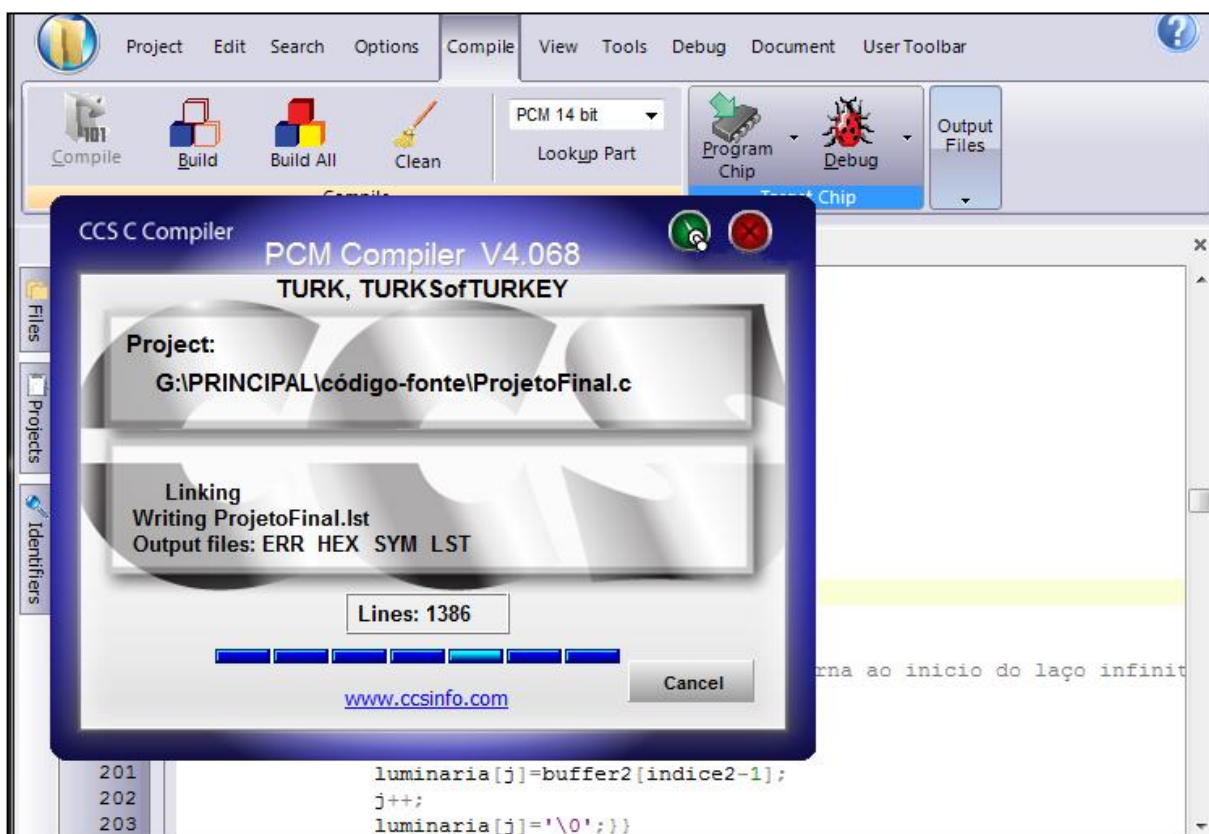


Figura 5.14 – Compilador do CCS C Compiler em execução. (AUTOR)

Com isso o compilador inicia a verificação da sintaxe do código fonte em busca de algum erro ou situação de perigo, paralelo a esta atividade são criadas oito novos arquivos com o mesmo nome, e com extensões diversos, sendo uma delas a ‘.hex’ que foi utilizada nos testes iniciais e na gravação no microcontrolador PIC.

A gravação no microcontrolador PIC é feita com auxílio do kit de gravação PICKit 2, as etapas de gravação seguem as descritas na seção – Utilização do Kit de Gravação PICKit 2 - do capítulo anterior.

#### 5.4 Montagem dos Circuitos nas Placas

Tanto o circuito da placa principal, como o circuito da placa de relés foram feitas em placas de fenolite cobreada de face única. Para elaboração do leiaute do circuito foi utilizado o programa Proteus ARES 7 Professional em conjunto com o programa Proteus ISIS 7 Professional.

Na Figura 5.15 é mostrado o desenho do circuito da placa de relés feita no Proteus ARES, e o resultado na placa de fenolite já descobreada, e pronta para a soldagem dos componentes.

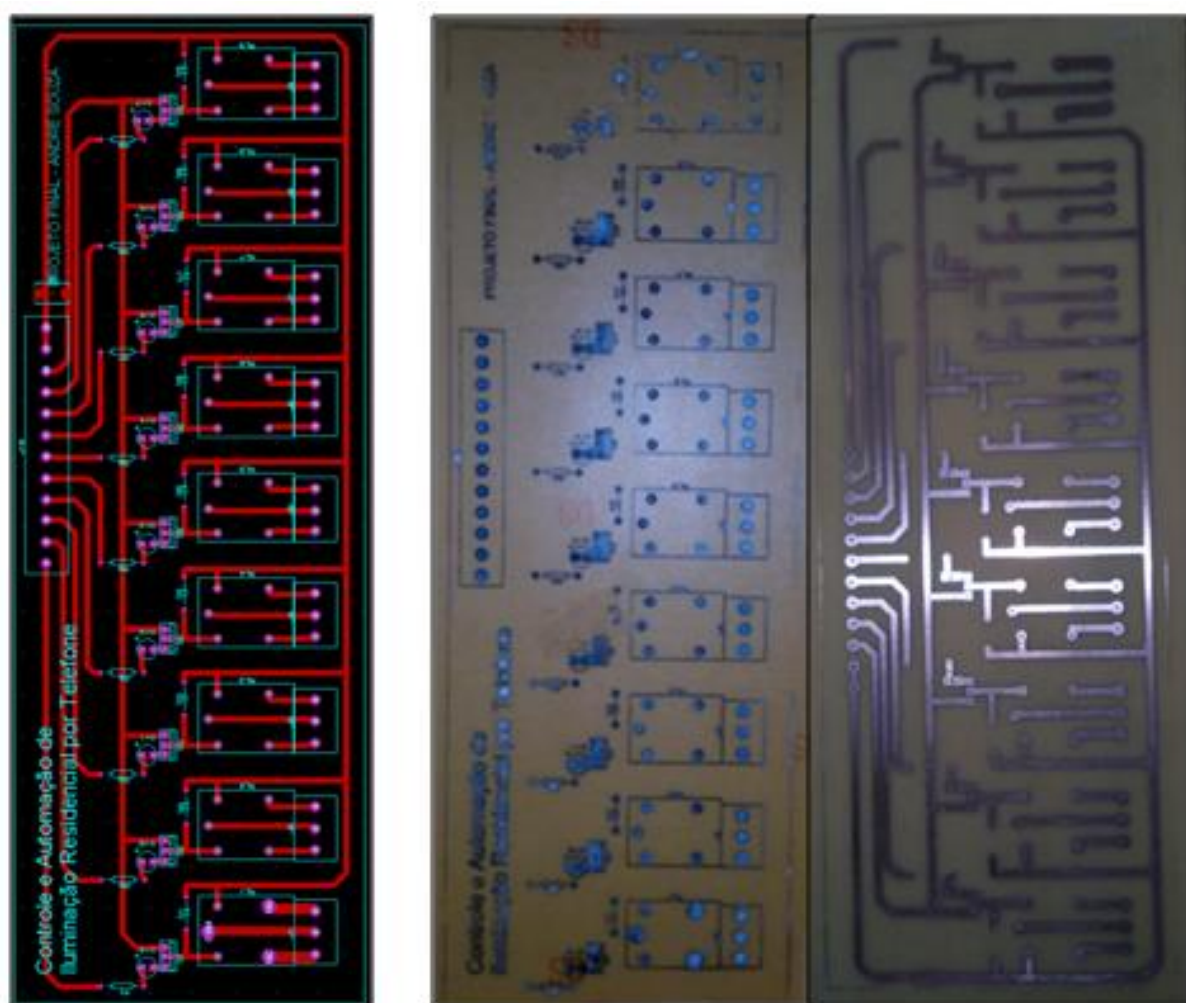


Figura 5.15 – Desenho da Placa de Relés e Placa descobreada pronta. (AUTOR)



Na Figura 5.16 é mostrada a disposição dos componentes eletrônicos da placa principal do sistema, com a descrição e legenda por meio de enumeração destes componentes.



Figura 5.16 – Placa Principal e Disposição dos Componentes. (AUTOR)

Na Figura 5.17 é mostrada a placa de relés na fase final e com os componentes eletrônicos já soldados. No primeiro relé foi utilizado um LED vermelho para diferenciar, por ser este o relé de controle do interruptor/dispositivo, os oito restantes foram utilizados para controle da iluminação dos cômodos da casa.

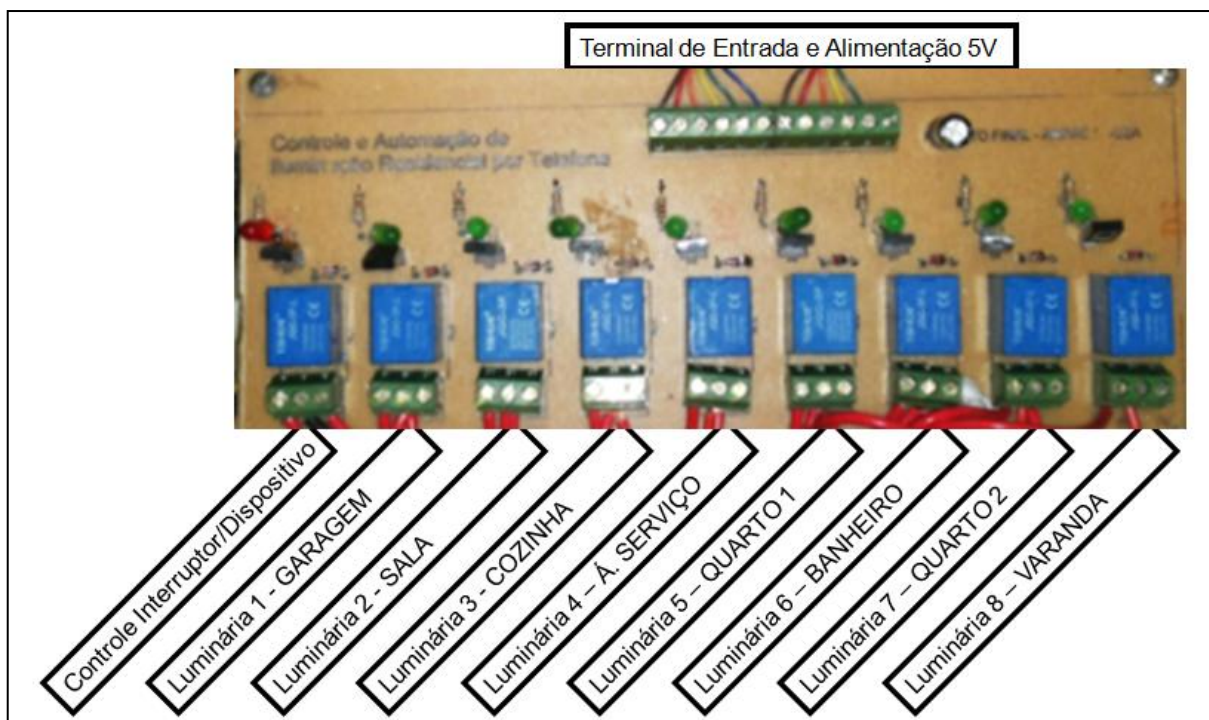


Figura 5.17 – Placa de Relés (AUTOR)

## 5.5 Montagem do Protótipo

A maquete foi confeccionada de acordo com disposição dos componentes previamente definidas, descrito na Figura 5.2. A casa foi confeccionada utilizando madeira aglomerada, fixada com a utilização de tiras de alumínio e parafusos com porcas de 2 mm de diâmetro, conforme é mostrada na Figura 5.18.



Figura 5.18 – Maquete da casa em fase de construção (AUTOR)

Após finalização da maquete da casa, foram fixados, em cada cômodo, bocais para lâmpadas incandescentes decorativas, que funcionam com tensão de 220 V e têm potência de 7 W. A escolha desta lâmpada deve-se ao seu tamanho reduzido. Na Figura 5.19 é mostrado o bocal e a lâmpada utilizados no projeto.



**Figura 5.19 – Bocal e lâmpada utilizados no projeto (AUTOR)**

Para elaboração da caixa de interruptores e tomadas, conforme ilustrado na Figura 5.20, foram utilizados:

- Dois conjuntos de 3 interruptores e um conjunto de 2 interruptores para iluminação e um interruptor para acionamento das tomadas;
- Um conjunto de 2 tomadas para o módulo GSM e para o dispositivo, que foram ligados através de fontes de alimentação;
- Entrada para ligação de cabo de alimentação.



**Figura 5.20 – Caixa de Interruptores e Tomadas. (AUTOR)**



A placa principal, placa de relés, o kit de desenvolvimento do módulo GSM em conjunto com sua antena e a caixa de interruptores e tomadas foram fixados de acordo com a disposição mostrada na Figura 5.21.

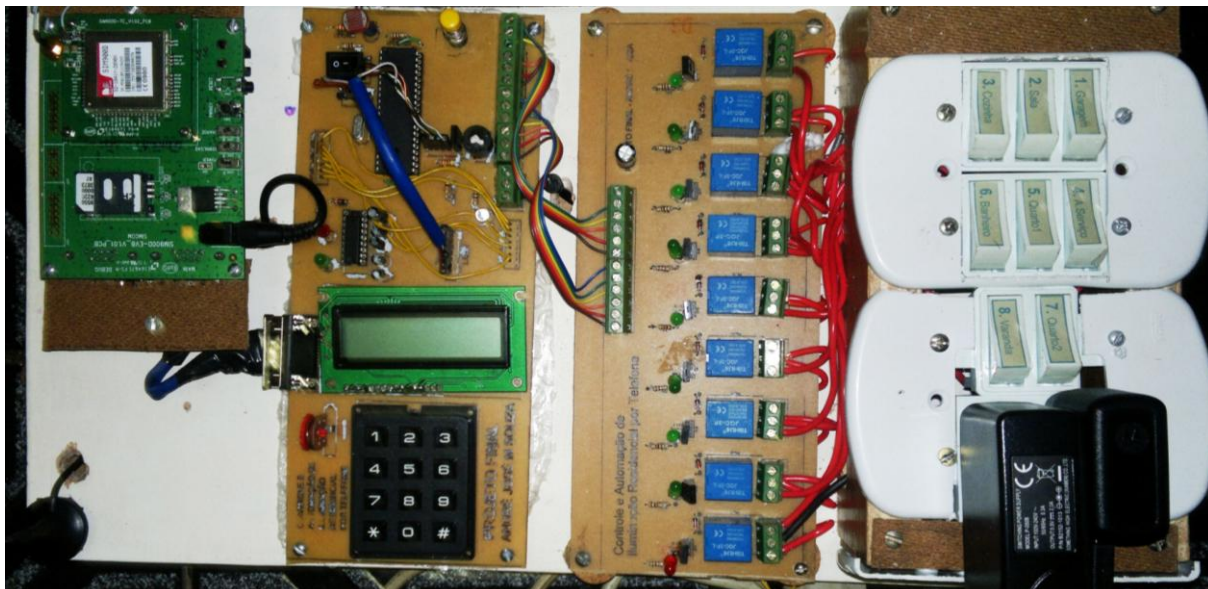


Figura 5.21 – Disposição dos componentes a frente da casa. (AUTOR)

Para melhoramento do leiaute do protótipo, as fiações como da antena do módulo, de fonte de alimentação do módulo e do dispositivo, e as fiações direcionadas às lâmpadas foram passadas por de baixo do tablado principal. As ligações referentes à corrente elétrica seguem no esquema mostrado na Figura 5.22.

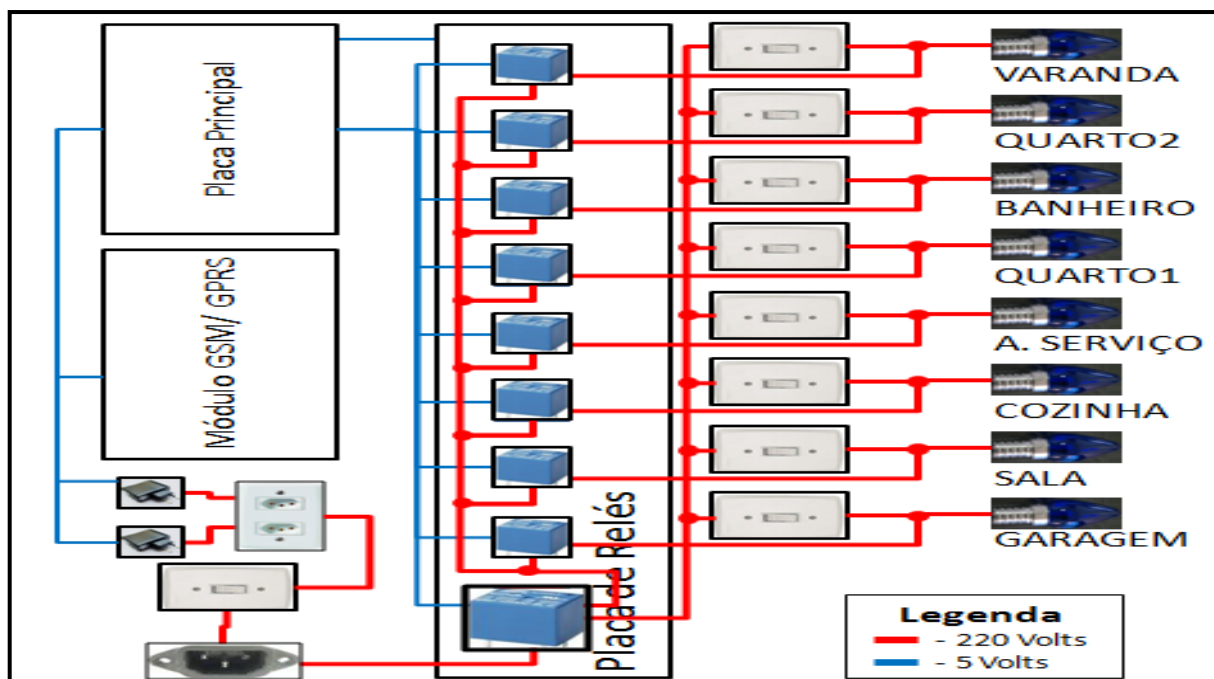


Figura 5.22 – Esquema das ligações elétricas. (AUTOR)

## CAPÍTULO 6 – RESULTADOS OBTIDOS

Neste capítulo são apresentados os resultados obtidos para chegar à solução do problema apresentado no Capítulo 2 – Apresentação do Problema – através da realização de simulações. São apresentados também alguns problemas encontrados.

### 6.1 Simulações

Esta fase do projeto teve como objetivo testar as funcionalidades propostas do dispositivo, simulando uma situação real. Antes de iniciar as simulações com a utilização do dispositivo proposto, o protótipo foi ligado para a realização de testes nos interruptores e tomadas, conforme é mostrado nas Figuras 6.1 e 6.2.

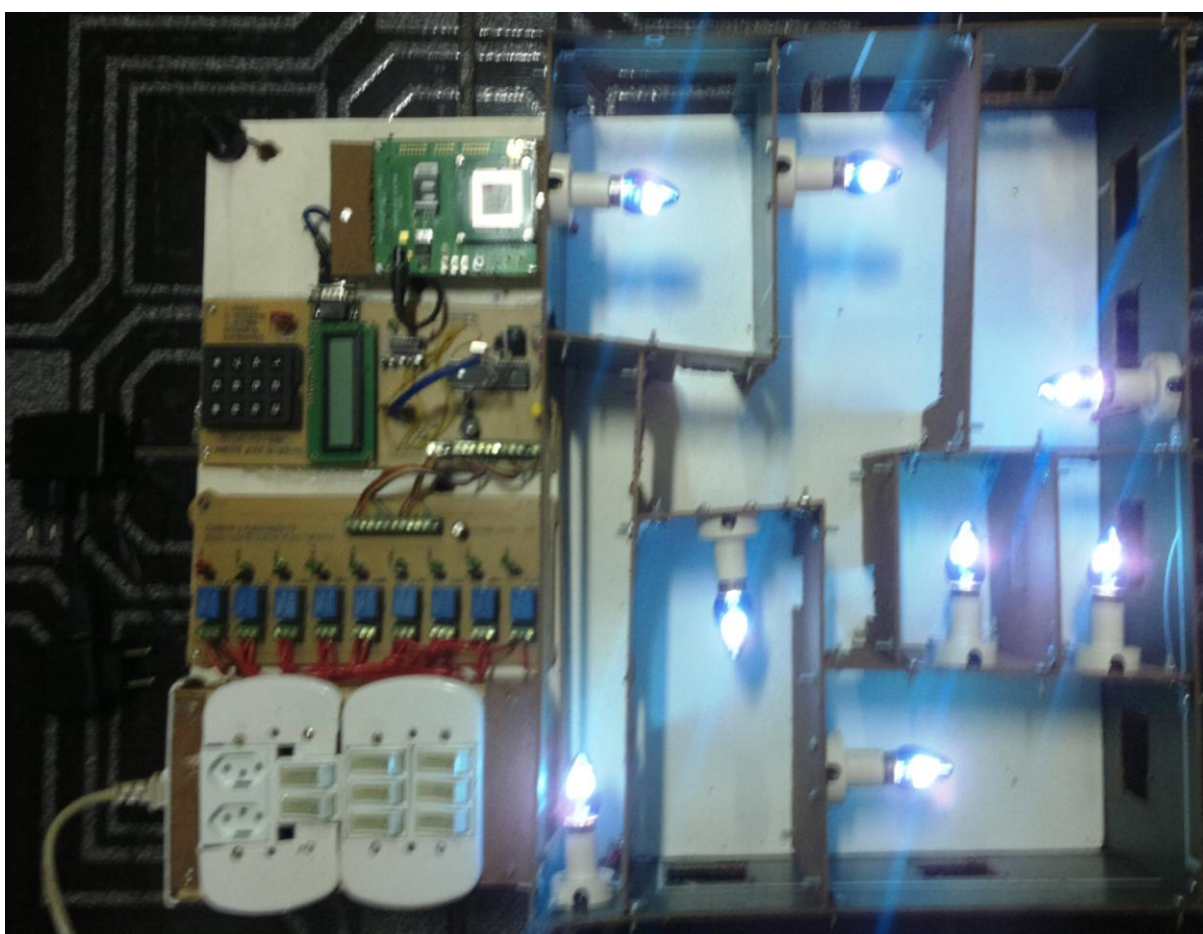
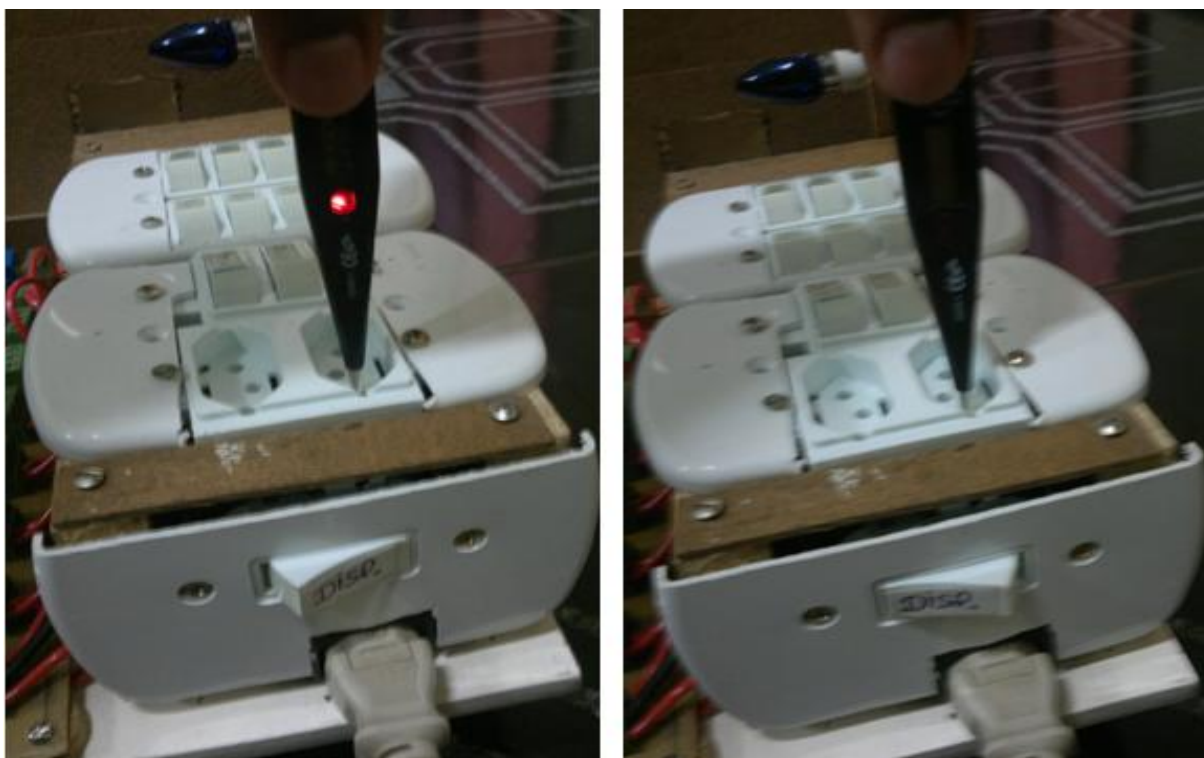


Figura 6.1 – Teste nos Interruptores (AUTOR)



**Figura 6.2 – Teste nas Tomadas com Chave de Teste. (AUTOR)**

Após testes de interruptores e tomadas, o dispositivo proposto e o módulo GSM foram religados nas tomadas, para início das simulações. As simulações realizadas tiveram como metas a obtenção dos seguintes resultados:

- Cadastro do número do cliente no sistema;
- Alteração do estado da iluminação da garagem e varanda pela utilização do sensor de luminosidade (LDR).
- Alteração do estado da iluminação da casa através da mensagem SMS enviada pelo cliente;

#### **6.1.1 Cadastro do número do cliente no sistema**

Esta simulação tem como finalidade observar o funcionamento do dispositivo ao ser ligado, até o envio da mensagem de instruções ao cliente. Este teste foi executado em torno de 20 vezes, de forma a comprovar que o dispositivo está realizando todas as etapas laborais do sistema. Os passos desta simulação foram:

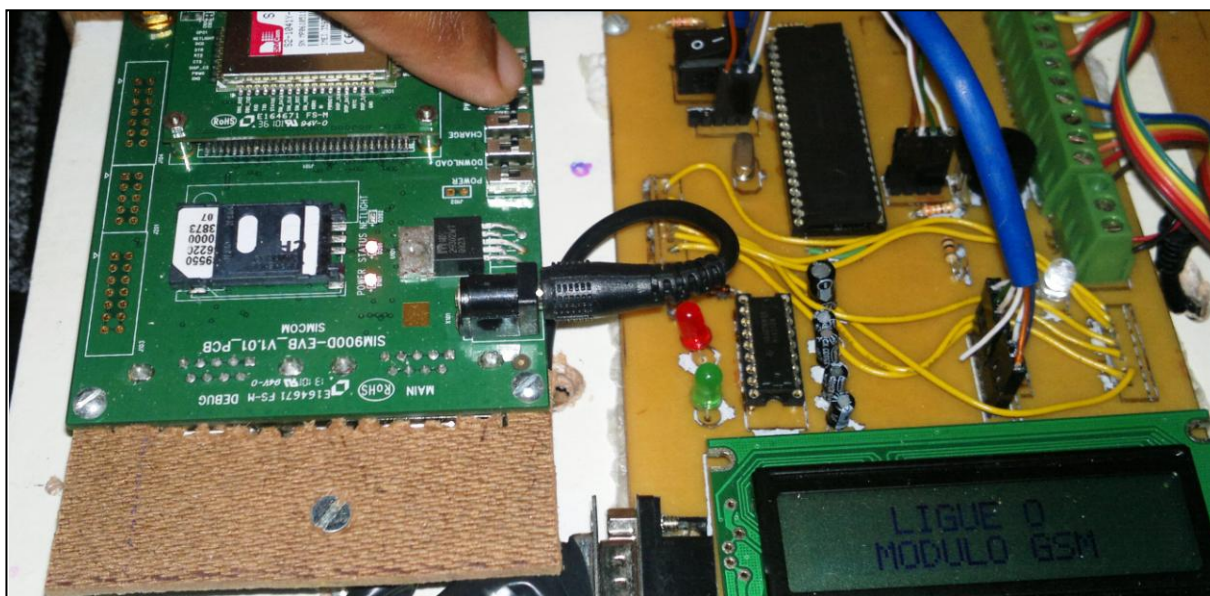
- Ligar o interruptor do dispositivo e do módulo GSM. Será emitido um bip, e uma mensagem será mostrada no módulo LCD, conforme é mostrado na Figura 6.3.





**Figura 6.3 – Protótipo – Mensagem Inicial do sistema (AUTOR)**

- Ao pressionar alguma tecla, será mostrada no módulo LCD a seguinte mensagem: “Ligue o módulo GSM”, antes de passar para próxima etapa é necessário ligar o módulo GSM, conforme é mostrado na Figura 6.4.



**Figura 6.4 – Protótipo – Ligando o módulo GSM. (AUTOR)**

- Após ligar o módulo GSM, pressione qualquer tecla, será realizada a limpeza das mensagens que posso estar armazenadas na memória do cartão SIM.

- No módulo LCD, será solicitada a digitação do número do cliente, conforme ilustra a Figura 6.5. Caso seja digitado o número incorretamente, a tecla '\*' tem como função apagar. Após digitar o número do cliente, a tecla '#' deve ser pressionada.

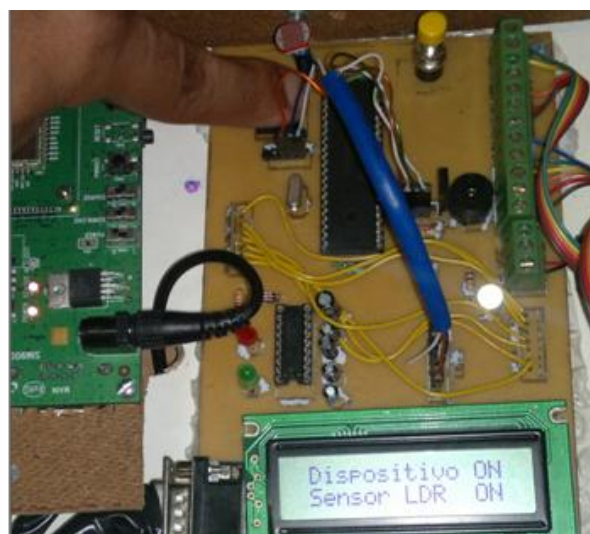


**Figura 6.5 – Protótipo – Digitando o número do cliente (AUTOR)**



**Figura 6.6 – Protótipo – Confirmando o número do cliente (AUTOR)**

- No módulo LCD, aparecerá o número digitado, conforme é mostrado na Figura 6.6. Para confirmar o cliente deve pressionar a tecla '#', caso o número não esteja correto, o cliente deve pressionar a tecla '\*', com isso o programa retorna ao passo anterior.
- A confirmar o número, a próxima etapa é o envio das duas mensagens SMS, a primeira com as instruções de uso e a segunda com a mensagem SMS padrão, que o cliente irá utilizar posteriormente. Ao finalizar o envio das mensagens são emitidos dois bips.



**Figura 6.7 – Protótipo – Mensagem final do sistema.  
Sensor LDR desligado (lado esquerdo) e Sensor LDR ligado (lado direito)**



- O último passo desta fase de testes será a visualização no módulo LCD das mensagens que são mostradas da Figura 6.7, que dependem se a chave do sensor LDR está ligada ou desligada.

Os resultados obtidos nesta primeira etapa do dispositivo demonstraram um funcionamento satisfatório. Nos testes realizados não ocorreram erros do sistema, foram utilizados dois números para a realização dos testes, sendo um da operadora B e outro da operadora C.

O tempo de envio da mensagem pelo sistema até o recebimento pelo cliente se mostrou curto na maioria dos testes realizados, sendo que em 2 ou 3 testes a mensagem demoram mais que 5 minutos para ser recebida pelo celular do cliente, possivelmente devido a algum problema relacionado as operadoras de telefonia.

#### 6.1.2 Alteração da iluminação pelo sensor de LDR

Esta simulação tem como objetivo observar a utilização do sensor LDR, que no caso deste trabalho tem como função o acionamento da iluminação da garagem e da varanda quando a iluminação natural estiver abaixo da estipulada pelo sistema. Para isso é necessário que a chave do sensor LDR esteja na posição ligada, conforme é destacado na Figura 6.8.



Figura 6.8 – Chave do Sensor LDR ligada. (AUTOR)

Foram realizados cerca de 20 testes utilizando uma cobertura preta sobre o sensor LDR, a fim de simular um ambiente com pouca luminosidade natural.

Todos os testes se mostram satisfatórios, com a colocação da cobertura, de acordo com o código fonte após três segundos o dispositivo teve realizar uma nova análise da tensão obtida pelo pino onde o sensor LDR foi interligado. Esta nova análise tem como objetivo que o dispositivo não seja acionado indevidamente, devido fatores normais que podem vir a acontecer, como um sombreamento sobre o sensor LDR, ou no caso em que o dispositivo já esteja ativo, alguma claridade repentina sobre o sensor.

Após a nova análise, com a baixa luminosidade natural, tem o acionamento da iluminação da garagem e da varanda, conforme estabelecido no código fonte tem o envio da mensagem SMS ao cliente e a emissão de dois bips. Com a luminosidade restabelecida, a iluminação da garagem e da varanda é desligada, é enviada uma mensagem SMS ao cliente e são emitidos dois bips.

A Figura 6.9 ilustra os testes feitos com a colocação da cobertura preta sobre o sensor LDR, acionamento da iluminação da varanda e da garagem e o envio da mensagem SMS ao cliente.



Figura 6.9 – Protótipo – Testes feitos para o sensor LDR (AUTOR)

### 6.1.3 Alteração da iluminação pelo cliente

Esta simulação tem como objetivo realizar testes no dispositivo através do envio da mensagem SMS padrão pelo cliente. Foram realizados diversos testes, que mostram satisfatoriamente o funcionamento para qual o dispositivo foi elaborado.

Do instante em que a mensagem SMS padrão é enviada pelo cliente até a alteração da iluminação que foi solicitada demora certa de 30 segundos, isto porque o sistema ao receber a mensagem, realiza a leitura da mensagem, verifica se o número do remetente é igual ao do registrado no sistema, e examina caractere por caractere da mensagem para que possa ser realizada a alteração solicitada.

Depois de realizada a alteração da iluminação, o sistema envia uma mensagem SMS ao cliente com relatório da iluminação que está ligada e desligada. Na Figura 6.10 são mostradas as três etapas da alteração da iluminação pelo cliente.



**Figura 6.10 – Alteração pelo cliente. Mensagem enviada pelo cliente (lado esquerdo), alteração no protótipo (no centro), mensagem recebida pelo cliente (lado direito) (AUTOR)**



## 6.2 Problemas Encontrados

Tinha-se como proposta inicial a alteração da iluminação através de ligação telefônica, para isso seria utilizado um circuito integrado de reconhecimento de tons DTMF, ou seja, ao pressionar uma tecla no telefone do cliente o sistema a identificaria com isso poderia relacionar a tecla com a iluminação desejada.

Esta proposta se tornou inviável devido à utilização do módulo GSM. Isto porque ao realizar uma chamada, os dados enviados pelo canal de voz, são digitalizados, codificados e enviados pelo sistema de transmissão digital. No módulo GSM o sinal é reconstruído, mas devido à compressão de voz utilizada, os tons DTMF não são recebidos corretamente.

Com isso foi realizada a alteração da proposta pelo uso de mensagem SMS, que é uma solução mais adequada e atual, além de ser mais precisa.

Um problema relatado nas simulações foi relacionado ao acionamento dos interruptores em conjunto com o dispositivo. Ocorre que, por exemplo, se deixarmos o interruptor da sala, cozinha e varanda ligadas, ao acionarmos através da mensagem SMS padrão apenas a iluminação da varanda, a iluminação da sala e cozinha também serão acionadas. Isso ocorre devido à corrente elétrica percorrer os interruptores que estiverem acionados.

Para solucionar este problema, os interruptores teriam que ser substituídos por interruptores sem retenção, ou com a utilização de relés de controle interruptor/dispositivo individuais para cada cômodo. Contudo não haveria tempo hábil para substituição dos interruptores ou montagem de outra placa de relés, ficando a proposta de troca dos interruptores como sugestão para trabalhos futuros.

Com esse problema, todos os testes realização do protótipo, os interruptores tinham que estar na posição desligado, para que não houvesse interferências que prejudicassem nas simulações.

## **CAPÍTULO 7 – CONSIDERAÇÕES FINAIS**

### **7.1 Conclusões**

Neste trabalho foi desenvolvido um protótipo de uma casa com o intuito de simular o real funcionamento do dispositivo. Para ser o mais fiel possível a um sistema real, os componentes utilizados no protótipo utilizam a tensão utilizada pela iluminação de uma casa real.

O presente projeto tem como foco a elaboração de um dispositivo de automação residencial de baixo custo, e que seja necessário apenas um celular comum que envie mensagens SMS para controlá-lo.

Apesar dos problemas que surgiram ao longo do percurso de elaboração deste trabalho, o objetivo do trabalho foi alcançado. O dispositivo é capaz de realizar o controle e automação da iluminação residencial através do envio da mensagem SMS padrão pelo cliente ou pela utilização do sensor de luminosidade.

Para realização deste projeto, foi necessário o desenvolvimento de duas placas: a placa principal e a placa de relés, com isso o autor pretendia que o dispositivo fosse de mais baixo custo possível.

No caso do módulo GSM utilizado também poderíamos ter realizado o desenvolvimento em placa de circuito, porém devido ao tempo reduzido o autor achou melhor realizar a compra do kit de desenvolvimento do módulo GSM completo, o que elevou o preço final para elaboração do projeto acadêmico, conforme pode ser observado no Apêndice B – Custos dos Componentes do Dispositivo.

É possível diminuir os gastos referentes ao envio das mensagens SMS, dependendo do plano da operadora utilizada, tendo, por exemplo, operadoras que cobram apenas uma taxa fixa ao dia para envio de mensagens SMS.

Concluindo, que os resultados obtidos cumpriram com as propostas e os objetivos planejados para este trabalho e o protótipo encontra-se em funcionamento e operando de acordo com a programação pretendida.

## 7.2 Propostas para Trabalhos Futuros

Controle e Automação Residencial é uma área muito ampla, que está em crescimento constante, como proposta para trabalhos futuros, os principais pontos verificados para a continuidade e melhorias deste projeto são:

- Avisar ao cliente em caso de falta de luz, para isso seria necessário a utilização de uma bateria externa, já que a queda de energia faria com que o dispositivo e o módulo GSM desligassem;
- Cadastro de outros números no sistema, em caso de alguma alteração, o dispositivo enviaria a mensagem SMS apenas ao número do cliente que solicitou a alteração.
- Ligar a iluminação utilizando interruptores eletrônicos, em conjunto com o controle e automação da iluminação pelo telefone, ou seja, tanto o acionamento via mensagem SMS ou pelo interruptor, o dispositivo seria o responsável pelo acionamento da iluminação.
- Realizar a 'leitura' da iluminação dos cômodos, com isso tanto o acionamento via interruptor, como uma lâmpada queimada seriam identificados pelo sistema.
- Utilização do projeto base, para acionamento de outros utensílios da casa, como: ar condicionado, cortinas, eletrodomésticos, etc.

## REFERÊNCIAS BIBLIOGRÁFICAS

ABREU, R. S. **Automação Residencial**: um pouco de história. Ribeirão Preto, 2003. Apresentação em PowerPoint. Arquivo baixado da Internet. Disponível em: <[http://www.aureside.org.br/publicacoes/download/automacao\\_residencial.zip](http://www.aureside.org.br/publicacoes/download/automacao_residencial.zip)>. Acesso em: 29 mar. 2011.

BERNAL, P. S. M. **Comunicações Móveis: Tecnologias e Aplicações**. 1ª. Ed. São Paulo: Érica, 2010.

BRAGA, N. C. **Tudo sobre relés**. Revista Saber Eletrônica. 2010? Disponível em: <<http://www.metaltex.com.br/tudosobredeles/tudo1.asp>> Acesso em 16 nov. 2011.

CAMPOS, D. A. L. **Sistema de Alarme de Intrusão utilizando o SMS**. Brasília: Faculdade de Tecnologia e Ciências Sociais Aplicada, UniCEUB, 2010.

DONO de loja de móveis usados é preso acusado de furtar residências. **CorreioWeb, Brasília**. Publicado em: 17 fev. 2011. Disponível em: <[http://www.correiobraziliense.com.br/app/noticia/cidades/2011/02/17/interna\\_cidadesdf,238337/dono-de-loja-de-moveis-usados-e-preso-acusado-de-furtar-residencias.shtml](http://www.correiobraziliense.com.br/app/noticia/cidades/2011/02/17/interna_cidadesdf,238337/dono-de-loja-de-moveis-usados-e-preso-acusado-de-furtar-residencias.shtml)> Acesso em: 07 mar. 2011.

FREITAS, L. **Automatizar para Economizar**. Lumière Eletric, São Paulo, edição 147, ano 12, jun. 2010. p. 36-41.

GIMENEZ, S. P. **Microcontroladores 8051**. 1ª ed. São Paulo: Pearson, 2005.

GSM World. **GSM**. 2011. Disponível em: <<http://www.gsm.org/technology/gsm/index.htm>> Acesso em: 30 abr. 2011.

HIGASHI, C. Tudo que o SIM Card pode fazer por você. **UOL Mobile - Tecnologia**. Publicado em: 02 mar. 2010. Disponível em: <<http://webinsider.uol.com.br/2010/03/02/tudo-que-o-sim-card-pode-fazer-por-voce/>> Acesso em 12 maio 2011.

LEAL, R. Viva na casa do futuro hoje. **Info Notícias**. Publicado em: 17 jan. 2011. Disponível em: <<http://info.abril.com.br/noticias/tecnologia-pessoal/viva-na-casa-do-futuro-hoje-17012011-2.shl>> Acesso em 29 mar. 2011.

ME COMPONENTES. **Módulo GSM/GPRS**, SIM900D. Disponível em: <<http://www.mecomp.com.br/sim900d.html>> Acesso em 09 set. 2011.

MICROCHIP TECHNOLOGY INC. **PIC16F87XA**. DataSheet: 28/40/44-Pin Enhanced Flash Microcontrollers. 2003. Disponível em: <<http://ww1.microchip.com/downloads/en/devicedoc/39582b.pdf>> Acesso em 9 set. 2011.

MURATORI, J. R. **Estamos preparados para a automação residencial?** Publicado em: [2010?]. Disponível em: <<http://www.forumdaconstrucao.com.br/conteudo.php?a=11&Cod=825>> Acesso em 30 mar. 2011.

NÚMERO de celulares no Brasil ultrapassa os 200 milhões. **Época Negócios**. Publicado em 19 jan. 2011. Disponível em: <<http://epocanegocios.globo.com/Revista/Common/0,,EMI203598-16418,00-NUMERO+DE+CELULARES+NO+BRASIL+ULTRAPASSA+OS+MILHOES.html>> Acesso em 27 abr. 2011.

PAIOTTI, R. **As dificuldades de trabalhar com Microcontroladores no Brasil**. Publicado em: 30 mar. 2009. Disponível para assinantes em: <<http://www.sabereletronica.com.br/secoes/leitura/1218>> Acessado em 30 mar. 2011.

PEREIRA, F. **Microcontroladores PIC: Programação em C**. 7ª. Ed. São Paulo: Érica, 2007.

ROMANO, R. A. **Iluminação: nossa aliada no conforto, relaxamento e economia**. Publicado em: 24 mar 2011. Disponível em: <[http://www.automatichouse.com.br/AutomaticHouse/WebSite/Informativo/iluminacao-nossa-aliada-no-conforto-relaxamento-e-economia,20110324115527\\_T\\_167.aspx](http://www.automatichouse.com.br/AutomaticHouse/WebSite/Informativo/iluminacao-nossa-aliada-no-conforto-relaxamento-e-economia,20110324115527_T_167.aspx)> Acesso em 30 mar. 2011.

SIMCOM. **Development Kit Manual**: SIM900D-EVB\_UGD\_V1.01. Publicado em: 01 jul. 2010. Disponível em: <[http://www.mecomp.com.br/rumo/SIM900D-EVB\\_UGD\\_V1.01.pdf](http://www.mecomp.com.br/rumo/SIM900D-EVB_UGD_V1.01.pdf)> Acesso em 09 set. 2011.

SOUZA, D. J. de; LAVINIA, N. C. **Conectando o PIC 16F877A: Recursos Avançados**. 2ª. Ed. São Paulo: Érica, 2005.

TEXAS INSTRUMENTS. **MAX232, MAX232I, DUAL EIA-232**. Datasheet do MAX232, 2004. Disponível em: <<http://www.ti.com/lit/ds/symlink/max232.pdf>> Acesso em 09 set. 2011.

THOMAZINI, D.; ALBUQUERQUE, P. U. B. de. **Sensores Industriais: Fundamentos e Aplicações**. 6ª. Ed. São Paulo: Érica, 2009

ZANCO, W. da S. **Microcontroladores PIC com Base no PIC16F877A: Técnicas de Software e Hardware para Projetos de Circuitos Eletrônicos**. 2ª. Ed. São Paulo: Érica, 2010.

## APÊNDICE A - CÓDIGO FONTE DO DISPOSITIVO

```

/*=====
PROJETO FINAL - Engenharia da Computação - UniCEUB
2o. Semestre de 2011

ANDRÉ JOSÉ MARTINS DE SOUZA
RA: 2051594/4

CONTROLE E AUTOMAÇÃO DE ILUMINAÇÃO RESIDENCIAL POR TELEFONE
=====*/

/*=====
DIRETIVAS DE PRÉ-COMPILAÇÃO
=====*/
#include <16f877a.h>           // Diretivas do PIC16F877A
#define adc=10                // Configuração do ADC
#define delay(clock=4000000)  // Informa a velocidade do clock
#define fuses xt, nowdt, put  // Fusíveis
#define fuses nolvp, nowrt, nocpd // Fusíveis
#define fuses nobrownout, noprotect // Fusíveis
#define use rs232 (baud=9600,xmit=pin_C6,rcv=pin_C7,restart_wdt,errors,stream=modem) // Ativa suporte comunicação serial RS232

#include <mod_lcd.c>           // Diretivas do display LCD
#include <string.h>            // Requisito para Comparar duas strings.
#define sensib 300            // Sensibilidade utilizada para o sensor LDR
#define LDR PIN_A1            // Liga e desliga o sensor LDR
#define LUM1 PIN_B0           // Luminária 1 - Garagem
#define LUM2 PIN_B1           // Luminária 2 - Sala
#define LUM3 PIN_B2           // Luminária 3 - Cozinha
#define LUM4 PIN_B3           // Luminária 4 - Área de Serviço
#define LUM5 PIN_B4           // Luminária 5 - Quarto 1
#define LUM6 PIN_B5           // Luminária 6 - Banheiro
#define LUM7 PIN_B6           // Luminária 7 - Quarto 2
#define LUM8 PIN_B7           // Luminária 8 - Varanda
#define LIN1 PIN_C0           // Teclado Matricial - pino 1 (linha 1)
#define LIN2 PIN_C1           // Teclado Matricial - pino 2 (linha 2)
#define LIN3 PIN_C2           // Teclado Matricial - pino 3 (linha 3)
#define LIN4 PIN_C3           // Teclado Matricial - pino 4 (linha 4)
#define LUM_todas PIN_C4      // Luminárias (Interruptor/Dispositivo)
#define BUZZER PIN_C5         // Alarme para uso geral
#define COL1 PIN_D0           // Teclado Matricial - pino 5 (coluna 1)
#define COL2 PIN_D1           // Teclado Matricial - pino 6 (coluna 2)
#define COL3 PIN_D2           // Teclado Matricial - pino 7 (coluna 3)

/*=====
CHAMADA INICIAL DAS FUNÇÕES BÁSICAS
=====*/
void Porta_Serial(void);      // Interrupção, recebendo comunicação serial
void limpa_GSM (void);        // Limpa mensagens SMS armazenadas no chip
void SMS_instrucoes1(void);    // Mensagem SMS, primeiras instruções

```

```

void SMS_instrucoes2(void);           // Mensagem SMS, estado das luminárias
void SMS_instrucoes3(void);           // Mensagem SMS, alteração pelo LDR, ligado
void SMS_instrucoes4(void);           // Mensagem SMS, alteração pelo LDR, desligado
char varre_teclas(void);              // Varre o tecla numérico
void SoaBuzzer(int w);                // Toca o buzzer.
int FuncaoDeParada(void);             // Varre teclas * e #, se *=0, se #=1

/*=====
                                     VARIÁVEIS
=====*/
long int    sensorLDR;
char        telefone[10], telefone2[10], luminaria[9], buffer1[74], buffer2[80];
int         i, j, tecla, indice1=0, indice2=0;
boolean     LDR_controle=0, LDR_controle1=0, testar=0;
boolean     iLUM1=0, iLUM2=0, iLUM3=0, iLUM4=0;
boolean     iLUM5=0, iLUM6=0, iLUM7=0, iLUM8=0;
boolean     LDR1=0, LDR0=0, tem_dado=0, SMS=0;

/*=====
                                     FUNÇÃO PRINCIPAL
=====*/
void main(){
    output_B (0);                     // Coloca nível em '0', as luminárias.
    output_low(LUM_todas);             // Coloca nível em '0', o relé int./disp.
    SoaBuzzer(1);                     // Buzzer de inicialização do sistema.
    lcd_ini();                         //Inicializa o display LCD.
    lcd_escreve("\f");
    lcd_escreve(" PROJETO FINAL ");
    lcd_escreve("\n");
    lcd_escreve("ANDRE M DE SOUZA");
    varre_teclas();                   // Cliente deve pressionar qualquer tecla para continuar
    delay_ms(350);
    lcd_escreve("\f");
    lcd_escreve("  LIGUE O  ");
    lcd_escreve("\n");
    lcd_escreve("MODULO GSM ");
    varre_teclas();                   // Cliente deve pressionar qualquer tecla para continuar
    delay_ms(50);
    limpa_GSM();
    setup_adc(adc_clock_internal);     // Configura o conversor AD interno
    setup_adc_ports(an0);              // Conf. as entradas do conversor AD interno
    delay_ms(100);
    while(testar==0){                 // Última linha de comando retorna '0' ou '1'
        telefone[0]='\0';
        i=0;
        lcd_ini();
        lcd_escreve("\f");
        lcd_escreve("DIGITE TELEFONE:");
        lcd_escreve("\n");
        delay_ms(100);
        do{
            tecla=varre_teclas();
            if(tecla!=26){              // TECLA # (ENTER)

```



```

        if(tecla==127 && i>0){                // Se tecla '*', delata último caractere.
            i--;
            lcd_escreve("\b");
            lcd_escreve(' ');
            lcd_escreve("\b");
        }else{                                // Adiciona tecla na string telefone e LCD
            telefone[i]=tecla;
            i++;
            lcd_escreve(tecla);}              // Fim do if (tecla != 26)
        telefone[i]='\0';
        delay_ms(350);
    }while(tecla!=26);
    lcd_escreve("\f");
    lcd_escreve("FONE: ");
    printf(lcd_escreve,"%s",telefone);
    lcd_escreve("\n");
    lcd_escreve("NAO(*) OU SIM(#)");
    testar=FuncaoDeParada();
}                                              // Fim do while(testar==0)
lcd_escreve("\f");
lcd_escreve("02 MENSAGENS SMS");
lcd_escreve("\n");
lcd_escreve("SERAO ENVIADAS..");
SMS_instrucoes1();
SoaBuzzer(2);
if(input(LDR)){ LDR0=0;}
else{          LDR0=1;}
output_high(LUM_todas);                      // Coloca nível em '1', o relé interruptor/dispositivo
limpa_GSM();
while(true){                                  // Loop INFINITO
    loop:                                     // goto loop, caso não seja o número cadastrado
    limpa_GSM();
    enable_interrupts(int_rda);               // Habilita interrupção de recepção de dados
    enable_interrupts(global);               // Habilita interrupção global
    tem_dado=0;
do{                                           // Loop PRINCIPAL, aguarda SMS ou Sensor LDR
    buffer1[0]='\0';                          // Limpa o buffer1
    buffer2[0]='\0';                          // Limpa o buffer2
    indice1=indice2=0;                       // Zera os contadores utilizados nos buffers
    set_adc_channel(0); delay_us(200);
    sensorLDR = read_adc();                  // Efetua conversão AD
    if(input(LDR)&&LDR0==0){                  // Mensagem LCD, quando sensor LDR ligado
        lcd_ini();
        lcd_escreve("\f");
        lcd_escreve(" Dispositivo ON ");
        lcd_escreve("\n");
        lcd_escreve(" Sensor LDR  ON ");
        LDR0=1;}
    if(!input(LDR)&&LDR0==1){                // Mensagem LCD, quando sensor LDR desligado
        lcd_ini();
        lcd_escreve("\f");
        lcd_escreve(" Dispositivo ON ");
        lcd_escreve("\n");
    }
}

```

```

    lcd_escreve(" Sensor LDR  OFF");
    LDR0=0;}
if(sensorLDR<sensib && LDR_controle==0 && input(LDR)){
    delay_ms(3000);                // Sensor LDR ligado, escureceu.
    sensorLDR = read_adc();        // Realiza novo teste, e seta a
    if(sensorLDR<sensib){          // variavel de controle
        LDR_controle1=1;}}
if(sensorLDR>=sensib && LDR_controle==1 && input(LDR)){
    delay_ms(3000);                // Sensor LDR ligado, claridade.
    sensorLDR = read_adc();        // Realiza novo teste, e seta a
    if(sensorLDR>=sensib){        // variavel de controle (Tem que
        LDR_controle1=1;}}        // estar ativo devido ao sensor LDR)
if(tem_dado==1){                  // Caso receba um caractere "Próxima Linha"
    delay_ms(1000);
    tem_dado=0;
    if(buffer1[4]=='T' && buffer1[5]=='I'){ // Verifica se é mensagem SMS
        SMS=1;}}
}while(!(SMS==1)&&!(input(LDR)&&LDR_controle1==1)); //Fim do DO.
//Sai do WHILE: recebeu uma mensagem SMS, ou variavel de controle LDR setada

if(SMS==1){
    SMS=0;
    buffer1[0]='\0';
    buffer2[0]='\0';
    indice1=0;
    indice2=0;
    printf("AT+CMGR=1\r");        // Realiza leitura da mensagem SMS
    delay_ms(3000);
    i=0; j=0;
    telefone2[0]='\0';
    for(indice1=0;indice1<=74;indice1++){
        if(buffer1[indice1]==' '){
            delay_ms(1); i++;
            if(i==1){              // Adiciona número do remetente na string telefone2
                telefone2[0]=buffer1[indice1+7];
                telefone2[1]=buffer1[indice1+8];
                telefone2[2]=buffer1[indice1+9];
                telefone2[3]=buffer1[indice1+10];
                telefone2[4]=buffer1[indice1+11];
                telefone2[5]=buffer1[indice1+12];
                telefone2[6]=buffer1[indice1+13];
                telefone2[7]=buffer1[indice1+14];
                telefone2[8]='\0';}}}
    if(strncmp(telefone,telefone2,8)!=0){ //Compara telefone e telefone2
        SoaBuzzer(1);
        goto loop;}              // Se diferente, retorna ao inicio do laço principal
    for(indice2=0;indice2<=80;indice2++){ // Leitura da mensagem SMS padrão
        if(buffer2[indice2]==' '){
            delay_ms(1);
            luminaria[j]=buffer2[indice2-1]; // Apenas os 'X' da mensagem SMS
            j++;
            luminaria[j]='\0';
        }
    }
} // Fim do if e do for.

```

```

if(luminaria[0]=='1' && LDR1==0){                                     // LUMINÁRIA 1 - GARAGEM
    output_high(LUM1);
    iLUM1=1;
}else{ if(luminaria[0]=='0' && LDR1==0){
    output_low(LUM1);
    iLUM1=0;}}

if(luminaria[1]=='1'){                                              // LUMINÁRIA 2 - SALA
    output_high(LUM2);
    iLUM2=1;
}else{ if(luminaria[1]=='0'){
    output_low(LUM2);
    iLUM2=0;}}

if(luminaria[2]=='1'){                                              // LUMINÁRIA 3 - COZINHA
    output_high(LUM3);
    iLUM3=1;
}else{ if(luminaria[2]=='0'){
    output_low(LUM3);
    iLUM3=0;}}

if(luminaria[3]=='1'){                                              // LUMINÁRIA 4 - ÁREA DE SERVIÇO
    output_high(LUM4);
    iLUM4=1;
}else{ if(luminaria[3]=='0'){
    output_low(LUM4);
    iLUM4=0;}}

if(luminaria[4]=='1'){                                              // LUMINÁRIA 5 - QUARTO 1
    output_high(LUM5);
    iLUM5=1;
}else{ if(luminaria[4]=='0'){
    output_low(LUM5);
    iLUM5=0;}}

if(luminaria[5]=='1'){                                              // LUMINÁRIA 6 - BANHEIRO
    output_high(LUM6);
    iLUM6=1;
}else{ if(luminaria[5]=='0'){
    output_low(LUM6);
    iLUM6=0;}}

if(luminaria[6]=='1'){                                              // LUMINÁRIA 7 - QUARTO 2
    output_high(LUM7);
    iLUM7=1;
}else{ if(luminaria[6]=='0'){
    output_low(LUM7);
    iLUM7=0;}}

if(luminaria[7]=='1' && LDR1==0){                                     // LUMINÁRIA 8 - VARANDA
    output_high(LUM8);
    iLUM8=1;
}else{ if(luminaria[7]=='0' && LDR1==0){
    output_low(LUM8);
    iLUM8=0;}}

disable_interrupts(int_rda | global);                             // Desabilita interrupções
SMS_instrucoes2();
limpa_GSM();
SoaBuzzer(2);
tem_dado=0;}

```

```

    if(input(LDR) && sensorLDR<sensib && LDR_controle==0){
//Se sensor LDR ligado, luminosidade baixa e já não estiver ativo
    disable_interrupts(int_rda);
    disable_interrupts(global);
    output_bit(LUM1,1);
    delay_ms(10);
    output_bit(LUM8,1);
    iLUM1=iLUM8=1;
    LDR1=1;
    SMS_instrucoes3();
    SoaBuzzer(2);
    LDR_controle=1;
    LDR_controle1=0;}

    if(input(LDR) && sensorLDR>=sensib && LDR_controle==1){
//Se sensor LDR ligado, luminosidade alta e já estiver ativo
    disable_interrupts(int_rda);
    disable_interrupts(global);
    output_bit(LUM1,0);
    delay_ms(10);
    output_bit(LUM8,0);
    iLUM1=iLUM8=0;
    LDR1=0;
    SMS_instrucoes4();
    SoaBuzzer(2);
    LDR_controle=0;
    LDR_controle1=0;}
}} // Fim do while(true), laço infinito.

```

```

/*=====
                                     FUNÇÕES BÁSICAS
=====*/
char varre_teclas(void){
    // Varre o tecla numérico
    int tecla=47;
    // Coloca o caractere '/' para controle
    while(tecla==47){
        // Permanece até que uma tecla seja apertada
        output_low(COL2);
        // desativa segunda coluna
        output_low(COL3);
        // desativa terceira coluna
        output_high(COL1);
        // ativa primeira coluna
        if(input(LIN1)) tecla=49;
        // TECLA 1
        if(input(LIN2)) tecla=52;
        // TECLA 4
        if(input(LIN3)) tecla=55;
        // TECLA 7
        if(input(LIN4)) tecla=127;
        // TECLA * (DELETE)
        output_low(COL1);
        // desativa primeira coluna
        output_high(COL2);
        // ativa segunda coluna
        if(input(LIN1)) tecla=50;
        // TECLA 2
        if(input(LIN2)) tecla=53;
        // TECLA 5
        if(input(LIN3)) tecla=56;
        // TECLA 8
        if(input(LIN4)) tecla=48;
        // TECLA 0
        output_low(COL2);
        // desativa segunda coluna
        output_high(COL3);
        // ativa terceira coluna
        if(input(LIN1)) tecla=51;
        // TECLA 3
        if(input(LIN2)) tecla=54;
        // TECLA 6
    }
}

```

```

        if(input(LIN3)) tecla=57;           // TECLA 9
        if(input(LIN4)) tecla=26;         // TECLA # (ENTER)
        delay_ms(10);}                    // Fim do while (tecla == 47)
    return(tecla);}

```

```

void SMS_instrucoes1(void){    // 2 mensagens SMS (instruções e SMS padrão)
    fprintf(modem,"AT\r");
    delay_ms(300);
    fprintf(modem,"AT+CMGF=1\r");
    delay_ms(300);
    fprintf(modem,"AT+CSCS=\"GSM\"\r");
    delay_ms(300);
    fprintf(modem,"AT+CMGS=\"");
    fprintf(modem,"%s",telefone);
    fprintf(modem,"\"\r");
    delay_ms(300);
    fprintf(modem,"PARA LIGAR/DESLIGAR A ILUMINAÇÃO, ENVIE A ");
    fprintf(modem,"PROXIMO MENSAGEM A 99788538, SUBSTITUINDO O 'X' POR ");
    fprintf(modem,"'1' PARA LIGADO E '0' PARA DESLIGADO.");
    delay_ms(300);
    putc(26);putc(26);putc(26);
    delay_ms(4500);
    fprintf(modem,"AT\r");
    delay_ms(300);
    fprintf(modem,"AT+CMGF=1\r");
    delay_ms(300);
    fprintf(modem,"AT+CSCS=\"GSM\"\r");
    delay_ms(300);
    fprintf(modem,"AT+CMGS=\"");
    fprintf(modem,"%s",telefone);
    fprintf(modem,"\"\r");
    delay_ms(300);
    fprintf(modem,"X. GARAGEM\rX. SALA\rX. COZINHA\rX. SERVIÇO\r");
    fprintf(modem,"X. QUARTO1\rX. BANHEIRO\rX. QUARTO2\rX. VARANDA\r");
    delay_ms(300);
    putc(26);putc(26);putc(26);}

```

```

void SMS_instrucoes2(void){    // Alteração pelo cliente, listagem.
    fprintf(modem,"AT\r");
    delay_ms(700);
    fprintf(modem,"AT+CMGF=1\r");
    delay_ms(700);
    fprintf(modem,"AT+CSCS=\"GSM\"\r");
    delay_ms(700);
    fprintf(modem,"AT+CMGS=\"");
    fprintf(modem,"%s",telefone);
    fprintf(modem,"\"\r");
    delay_ms(700);
    fprintf(modem,"ALTERAÇÃO DA ILUMINAÇÃO PELO CLIENTE, LIGADAS:\r");
    if(iLUM1==1)    fprintf(modem," GARAGEM\r");
    if(iLUM2==1)    fprintf(modem," SALA\r");

```

```

if(iLUM3==1)    fprintf(modem," COZINHA\r");
if(iLUM4==1)    fprintf(modem," SERVIÇO\r");
if(iLUM5==1)    fprintf(modem," QUARTO1\r");
if(iLUM6==1)    fprintf(modem," BANHEIRO\r");
if(iLUM7==1)    fprintf(modem," QUARTO2\r");
if(iLUM8==1)    fprintf(modem," VARANDA\r");
fprintf(modem,"DESLIGADAS:\r");
if(iLUM1==0)    fprintf(modem," GARAGEM\r");
if(iLUM2==0)    fprintf(modem," SALA\r");
if(iLUM3==0)    fprintf(modem," COZINHA\r");
if(iLUM4==0)    fprintf(modem," SERVIÇO\r");
if(iLUM5==0)    fprintf(modem," QUARTO1\r");
if(iLUM6==0)    fprintf(modem," BANHEIRO\r");
if(iLUM7==0)    fprintf(modem," QUARTO2\r");
if(iLUM8==0)    fprintf(modem," VARANDA\r");
delay_ms(700);
putc(26);putc(26);putc(26); }

```

```

void SMS_instrucoes3(void){    // Alteração pelo sensor LDR, ligado.
    fprintf(modem,"AT\r");
    delay_ms(700);
    fprintf(modem,"AT+CMGF=1\r");
    delay_ms(700);
    fprintf(modem,"AT+CSCS=\"GSM\"\r");
    delay_ms(700);
    fprintf(modem,"AT+CMGS=\"");
    fprintf(modem,"%s",telefone);
    fprintf(modem,"\"\r");
    delay_ms(700);
    fprintf(modem,"ALTERAÇÃO DA ILUMINAÇÃO PELO SENSOR LDR:\r");
    fprintf(modem,"ILUMINAÇÃO DA GARAGEM E ")
    fprintf(modem,"DA VARANDA FORAM LIGADAS..");
    delay_ms(700);
    putc(26);putc(26);putc(26);}

```

```

void SMS_instrucoes4(void){    // Alteração pelo sensor LDR, desligado.
    fprintf(modem,"AT\r");
    delay_ms(700);
    fprintf(modem,"AT+CMGF=1\r");
    delay_ms(700);
    fprintf(modem,"AT+CSCS=\"GSM\"\r");
    delay_ms(700);
    fprintf(modem,"AT+CMGS=\"");
    fprintf(modem,"%s",telefone);
    fprintf(modem,"\"\r");
    delay_ms(700);
    fprintf(modem,"ALTERAÇÃO DA ILUMINAÇÃO PELO SENSOR LDR:\r");
    fprintf(modem,"ILUMINAÇÃO DA GARAGEM E ")
    fprintf(modem,"DA VARANDA FORAM DESLIGADAS..");
    delay_ms(700);
    putc(26);putc(26);putc(26);}

```

```

void SoaBuzzer(int w) {                                // Toca o buzzer.
    int i;
    for(i=1;i<=w;i++) {
        output_high(BUZZER);
        delay_ms(400);
        output_low(BUZZER);
        delay_ms(400);} }

int FuncaoDeParada(void){                             // Só continua, após pressionada a tecla '*' ou '#'
    int j=0;
    while(true){
        output_high(COL1);
        output_low(COL3);
        if(input(LIN4)){
            j=0;
            break;}
        output_low(COL1);
        output_high(COL3);
        if(input(LIN4)){
            j=1;
            break;}
        delay_ms(12);}
    return(j);    }                                // Retorna valor 0 se '*' e valor 1 se '#'

void limpa_GSM(void){                                  // Limpa SMS's da memória do cartão SIM
    fprintf(modem,"AT\r");
    delay_ms(700);
    fprintf(modem,"AT+CMGDA=\"DEL ALL\"\r");
    delay_ms(700);
    fprintf(modem,"AT+CMGDA=6\r");
    delay_ms(700); }

#int_rda        // Interrupção de recepção de dados
void Porta_Serial(void){
    char dado;
    dado=getc();
    if(indice1<74){
        buffer1[indice1]=dado;
        indice1++;}
    else{
        buffer2[indice2]=dado;
        indice2++;}
    if(dado==10){                                     // Tabela ASCII, representa Line Feed (Próxima Linha)
        tem_dado=1;} }

```

**APÊNDICE B – CUSTO DOS COMPONENTES DO DISPOSITIVO**

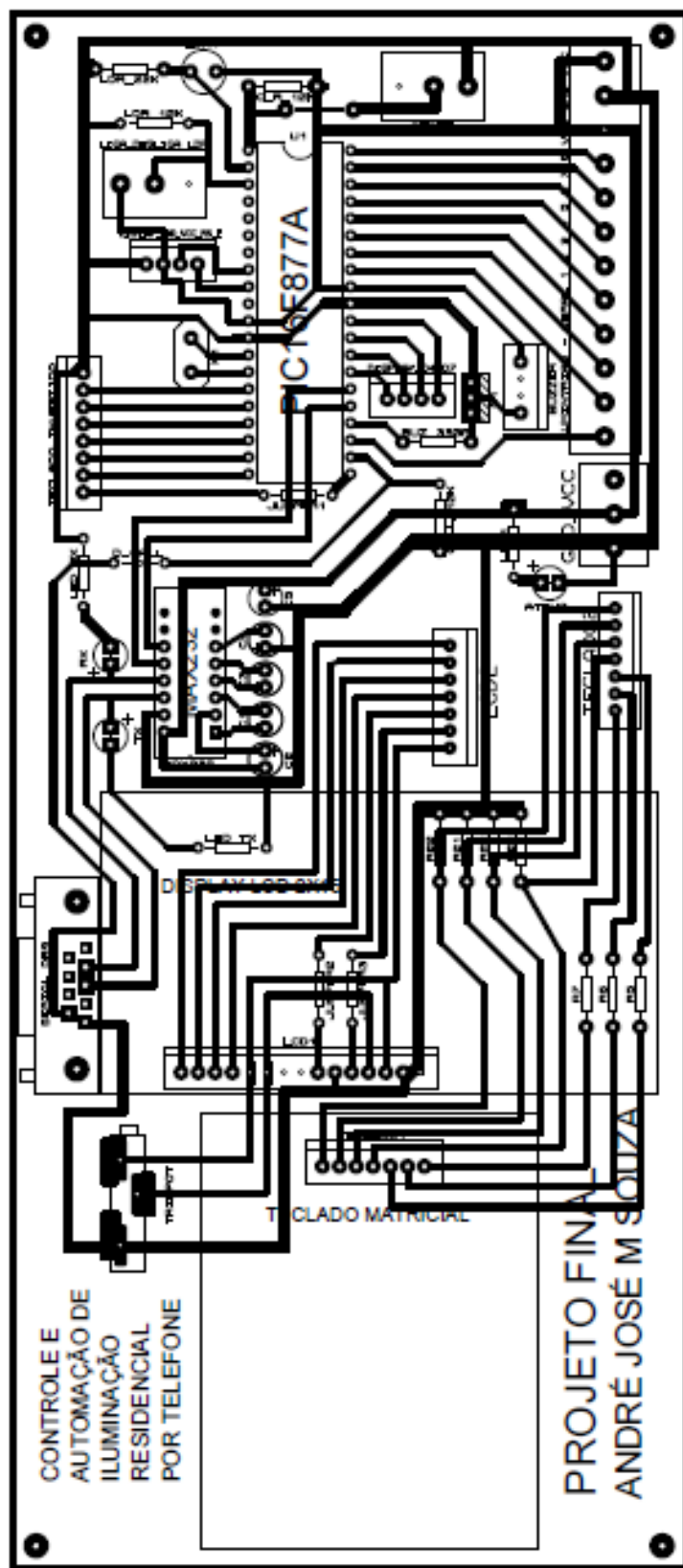
<b>Itens</b>	<b>Quantidade</b>	<b>Valor</b>
Kit de desenvolvimento para módulo SIM900D	1	R\$ 330,60
Teclado Numérico	1	R\$ 25,00
Módulo LCD 2x16	1	R\$ 20,00
Relés 5VDC	9	R\$ 19,80
Microcontrolador PIC16F877A	1	R\$ 15,00
Borne de 3 terminais	18	R\$ 14,40
Transistor NPN	10	R\$ 13,00
Placa de Fenolite 10cm X 30cm	2	R\$ 10,80
Chave Push-Button	1	R\$ 6,50
LED`s (verde, vermelho e branco)	12	R\$ 6,00
Fonte de Alimentação com saída USB	1	R\$ 5,00
Chave Interruptor	1	R\$ 3,50
Circuito Integrado MAX232	1	R\$ 2,50
Sensor LDR	1	R\$ 2,50
Resistores	24	R\$ 2,40
Capacitores	6	R\$ 2,00
Conector DB-9 macho	1	R\$ 2,00
Diodos	9	R\$ 1,80
Conector DB-9 fêmea	2	R\$ 1,30
Oscilador de Cristal 4 MHz	1	R\$ 1,30
Soquete para CI de 18 pinos	1	R\$ 1,30
Soquete para CI de 40 pinos	1	R\$ 1,30
Buzzer	1	R\$ 1,00
Trimpot	1	R\$ 1,00
<b>TOTAL</b>	<b>R\$ 490,00</b>	

FONTE: AUTOR



## APÊNDICE C – CIRCUITOS DAS PLACAS DO DISPOSITIVO

## Placa Principal



## Placa de Relés

